



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-03031759v2

HAL Id: hal-03031759

<https://hal.science/hal-03031759v2>

Submitted on 1 Apr 2022 (v2), 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** STIC-AmSud project GALOP.

39 *Cláudio Carvalho*: FUNCAP Pronem 4543945/2016.

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

41 *Raul Lopes*: FUNCAP Pronem 4543945/2016, French Agence Nationale de la Recherche under
42 contract ASSK ANR-18-CE40-0025-01. Work partially done in Departamento de Computação,
43 Universidade Federal do Ceará, with a PhD scholarship by FUNCAP.

44 *Ana Karolinna Maia*: FUNCAP Pronem 4543945/2016.

45 *Nicolas Nisse*: French Agence Nationale de la Recherche under contract Digraphs ANR-19-CE48-



© 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. 23:1–23:14

Leibniz International Proceedings in Informatics



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

46 0013-01.

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

48 **1 Introduction**

49 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
 50 to e as vw . The *in-degree* of a set $X \subseteq V(D)$, denoted by $d_D^-(X)$, is the number of arcs
 51 with tail outside of X and head inside of X . If $X = \{v\}$, we simply write $d_D^-(v)$ instead of
 52 $d_D^-(\{v\})$ (omitting braces). For an integer n , we denote by $[n]$ the set $\{1, \dots, n\}$.

53 A *network* $\mathcal{N} = (D, u)$ is formed by a digraph $D = (V, A)$ with a *capacity function*
 54 $u : A(D) \rightarrow \mathbb{Z}_+$. If all arcs in D have capacity λ , we simply write $u \equiv \lambda$. A *flow* in \mathcal{N} is a
 55 function $f : A(D) \rightarrow \mathbb{Z}_+$ such that $f(vw) \leq u(vw)$, $\forall vw \in A(D)$. For a vertex $v \in V(D)$,
 56 we define $f^+(v) = \sum_{vw \in A} f(vw)$ and $f^-(v) = \sum_{wv \in A} f(wv)$, that is, $f^+(v)$ and $f^-(v)$ are
 57 the amount of flow *leaving* and *entering* v , respectively. The *balance vector* of a flow f in a
 58 network $\mathcal{N} = (D, u)$ is the function $b_f : V(D) \rightarrow \mathbb{Z}$ associating each vertex $v \in V(D)$ to the
 59 value $f^+(v) - f^-(v)$. If $b_f(v) = \ell$, we may also say that v has *balance* ℓ . A vertex $s \in V(D)$
 60 is a *source* of f if $f^+(s) > 0$ and $f^-(s) = 0$. An (s, t) -*cut* on a network $\mathcal{N} = (D, c)$ is a set
 61 of arcs of the form (S, \bar{S}) , where S, \bar{S} form a partition of $V(D)$ such that $s \in S$ and $t \in \bar{S}$.
 62 The capacity of an (s, t) -cut is the sum of the capacities of arcs with tail in S and head in \bar{S}
 63 and it is denoted by $u(S, \bar{S})$. For other concepts on graphs and digraphs, we follow standard
 64 terminology as in [3, 6].

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

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

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

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

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

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

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

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

112 An s -branching flow in a network $\mathcal{N} = (D, u)$ is a flow f such that $b_f(s) = n - 1$ and
 113 $b_f(v) = -1$ for every $v \in V(D) \setminus \{s\}$. In other words, f reaches all vertices of D and each
 114 vertex other than s retains one unit of flow. Finding one s -branching flow in a given network
 115 is easy: since $\sum_{v \in V(D)} b_f(v) = 0$, we can reduce this problem to the problem of finding one
 116 (s, t) -flow, as discussed above. We can also find k arc-disjoint s -branching flows in polynomial
 117 time when $u \equiv n - 1$. In [2] it was shown that, in this case, \mathcal{N} admits an s -branching flow if
 118 and only if D contains an s -branching. Thus, applying Theorem 1, the authors provided a
 119 characterization of networks admitting k arc-disjoint s -branching flows, which are exactly the
 120 networks constructed on digraphs containing k arc-disjoint s -branchings. They also provided
 121 a polynomial time algorithm that finds such flows if they exist. We now discuss how the
 122 tractability of this problem, in general, depends on the choice of the capacity function.

123 In [2], the authors showed that the problem of deciding if a network has k arc-disjoint
 124 s -branching flows is NP-complete if every arc has capacity at most two, and in [4] this result
 125 was extended to networks with capacity at most ℓ , for every fixed $\ell \geq 2$. For most choices of
 126 larger capacities, the problem remains hard. In [4] it was shown that, unless the *Exponential*
 127 *Time Hypothesis* [13] fails, there is no polynomial time algorithm for the problem of finding
 128 k arc-disjoint s -branching flows in a network $\mathcal{N} = (D, u)$ with $u \equiv \lambda$ for any choice of λ
 129 such that $n/2 \leq \lambda \leq n - (\log n)^{1+\varepsilon}$, $\varepsilon > 0$, even if D is acyclic. In [7] the authors adapted
 130 this last proof to show that the same holds if $(\log n)^{1+\varepsilon} \leq \lambda \leq n/2$. The last two results
 131 have an intersection point, which is when $u \equiv n/2$. The two constructions result in the same
 132 network in this case, and it is worth noticing that it gives a polynomial time reduction. So,
 133 for $u \equiv n/2$, the problem is NP-hard. On the positive side, in [4] it was also shown that
 134 the problem is solvable in polynomial time when $\lambda = n - c$, for fixed k and $c \geq 1$. More
 135 precisely, the authors showed that the problem is XP for fixed k and c when the goal is to

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

23:4 A study of an Edmonds' like property to arc-disjoint branching flows

136 find k arc-disjoint s -branching flows. This was later improved to an FPT algorithm with the
 137 same parameter in [5].

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

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

$$147 \quad d_D^-(X) \geq k \left\lceil \frac{|X|}{\lambda} \right\rceil. \quad (1)$$

148
 149 If \mathcal{N} is a network constructed on such a digraph, we may also say that \mathcal{N} is (k, λ, s) -sufficient.

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

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

159 In Section 2, we show how to construct a family of a (k, λ, s) -sufficient networks not containing
 160 k arc-disjoint s -branching flows. On the positive side, we show that some positive cases of
 161 sufficiency do occur depending on the structure of D and on the choice of k and λ .

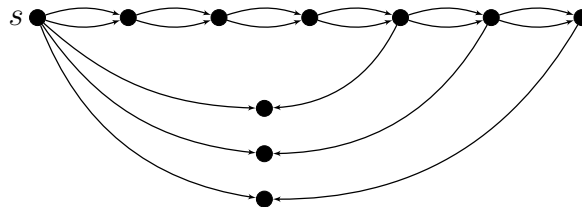
162 In Section 3, we show that (k, λ, s) -sufficiency guarantees the existence of k arc-disjoint
 163 s -branching flows in a network $\mathcal{N} = (D, u)$ when D is an s -branching where parallel arcs are
 164 allowed, and when D is a collection of internally disjoint multi-paths starting at s and all
 165 ending on the same vertex $t \in V(D)$. In other words, the flows are guaranteed to exist when
 166 the underlying simple graph of D is a tree or a spindle, and thus the considered property
 167 characterizes the existence of the flows in those classes of networks. Additionally, we show
 168 that we also have this guarantee in general digraphs when $k = 1$, when $\lambda = 1$, and when
 169 $\lambda = |V(D)| - 2$. Finally, we show that, for a network $\mathcal{N} = (D, u)$ with $u \equiv \lambda$ and an integer
 170 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
 171 arc-disjoint s -branching flows.

172 In Section 4, we show that the analyzed property is not enough to improve the tractability
 173 of the problem of finding the desired flows. Namely, we show that it is NP-complete to decide
 174 if a network \mathcal{N} admits k arc-disjoint s -branching flows even if its underlying simple graph is
 175 constructed by appending a path to the end of a spindle and we know beforehand that \mathcal{N}
 176 is (k, λ, s) -sufficient. Since this property can be easily verified in this class and it is always
 177 necessary for the existence of the flows, it is no surprise that it is not enough to guarantee
 178 the existence of the k arc-disjoint s -branching flows.

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

182 **2 Networks on (k, λ, s) -sufficient digraphs not containing the flows**

183 In Figure 1 we give an example of a $(2, n - 3, s)$ -sufficient network that does not contain 2
 184 arc-disjoint s -branching flows, and we remark that it is easy to generalise this example to
 185 $(2, n - c, s)$ -sufficient networks, for every fixed $c \geq 3$, and thus the best that we can expect is
 186 that every $(2, n - 2, s)$ -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).

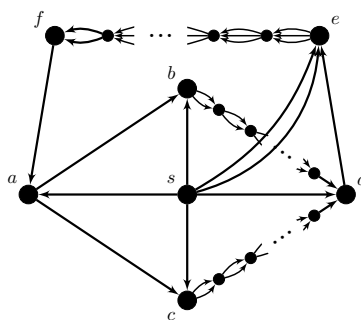


187 **Figure 1** Example of a network on a $(2, n - 3, s)$ -sufficient digraph that does not contain 2
 188 arc-disjoint s -branching flows. Here every arc has capacity $n - 3$.

187 Next, we show how to construct a family of (k, λ, s) -sufficient networks that do not contain
 188 k arc-disjoint s -branching flows.
 189

190 **Theorem 3.** For all $\lambda \geq 2$ and for all even $k \geq 2$, there exists a (k, λ, s) -sufficient network
 191 $\mathcal{N} = (D, u)$, with $u \equiv \lambda$, that does not admit k arc-disjoint s -branching flows.

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



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

201 Here, we say that a set $X \subseteq V(D)$ is *satisfied* if $d_D^-(X) \geq k \lceil |X|/\lambda \rceil$.
 202 To prove that D is (k, λ, s) -sufficient, we consider the case where $k = 2$ because for a
 203 larger k we multiply the in-degree of each subset of $V(D) - s$ by $k/2$. Observe that, except
 204 for s , every vertex has in-degree at least 2 in D and that for every $X \subseteq V(D) - s$ each vertex
 205 from $\{a, b, c, d\}$ which belongs to X sums at least one unit to its in-degree and the vertex e

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

206 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
 207 necessarily contains a, d, P and either B or C (or both), but its also true that $|X| \geq 2\lambda + 1$,
 208 which means that should be at least one extra arc entering in X for it to be satisfied. To
 209 verify the existence of this extra arc assume without loss of generality that $B \subset X$ (the
 210 case where $C \subset X$ is symmetric) and take the longest (z, d) -path in $D[X]$ such that $z \in C$.
 211 If no such path exists, there is at least one arc going from C to d which was not counted
 212 before and if the path exists, then either $z = c$ or the two arcs yz enters X , where y is the
 213 in-neighbor of z which, by the choice of z , cannot be in X .

214 Now consider that $D[X]$ is acyclic. Note that each source in $D[X]$ contributes with at
 215 least two units in the in-degree of X . Since each weak component of $D[X]$ has at least
 216 one source, then every X with $|X| \leq \lambda$ is satisfied. Similarly, the same also holds when
 217 $\lambda < |X| \leq 2\lambda$ if $D[X]$ has more then one weak component or only one component with many
 218 sources. Otherwise $D[X]$ has only one weak component W with a single source w , and we
 219 have 5 cases:

- 220 ■ $w \in B$. Observe that in this case $\{d, e\} \subset X$, once that $|X| \geq \lambda + 1$ and $|B| = \lambda - 1$,
 221 which means that $d_D^-(X) \geq 5$.
- 222 ■ $w \in C$. It's analogous to the previous one.
- 223 ■ $w = a$. In this case, there are at least two vertices from $\{b, c, d\}$ in X . That is, $d^-(X) \geq 4$.
- 224 ■ $w = d$. If d is the source, then $e \in X$, and thus $d_D^-(X) \geq 5$.
- 225 ■ $w \in V(P)$. Except for f , every vertex of P has in-degree three and since X must contains
 226 a , then $d_D^-(X) \geq 4$. When $w = f$, besides a , X must also contains at least one vertex
 227 from $\{b, c\}$ which guaranties $d_D^-(X) \geq 4$.

228 When $2\lambda + 1 \leq |X| \leq 3\lambda$, the in-degree of X must be at least 6. We start with the case
 229 that $D[X]$ has just one weak component. If w is a source in $D[X]$, again we have 5 subcases:

- 230 ■ $w \in B$. In this case, since $|B \cup \{d\}| = \lambda$, either there is a vertex in C which is also a source
 231 in $D[X]$ or $a \in X$. In both situations, $d, e \in X$ because $|B \cup C| < 2\lambda$. Thus, $d_D^-(X) \geq 6$.
- 232 ■ $w \in C$. It's analogous to the previous one.
- 233 ■ $w = a$. Once $|\{a, d\} \cup B \cup C| = 2\lambda$, if a is a source, there is a path in $D[X]$ that starts in
 234 a , goes through B or C , d and ends in some vertex of P .
- 235 ■ $w = d$. When d is a source, for sure $\{a, b, c, d, e\} \subset X$.
- 236 ■ $w \in V(P)$. Here, in order to avoid cycles, we know that if $w = f$, necessarily a, b, c and d
 237 are also included in X and if $w \neq f$, at least a, b and c are included in X but, in this
 238 case, w already contributes with 3 for the in-degree of X . Thus $d_D^-(X) \geq 6$.

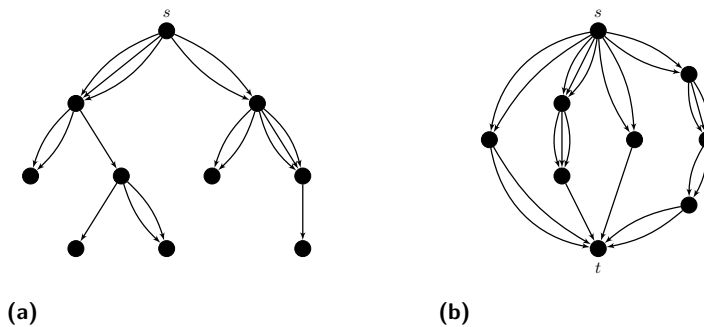
239 If $D[X]$ has more than 2 weak components we already know that $d_D^-(X) \geq 6$. If $D[X]$
 240 has two weak components Y, Z , one of them, say Y , has at least $\lambda + 1$ vertices and hence, as
 241 in the previous cases, $d_D^-(Y) \geq 4$. Thus, $d^-(X) \geq 6$.

242 Notice that if there are k arc-disjoint s -branching flows x_1, x_2, \dots, x_k in \mathcal{N} , as $d_D^-(B) =$
 243 $d_D^-(C) = k$, each one of these k arcs has to be used by a different flow, and carry at least
 244 $\lambda - 1$ units of flow, that is, one unit for each vertex of B and C . Now observe that $k/2$ of the
 245 arcs which enter in B come from a and the same for the ones that enter C , and then a must
 246 receive λ units of flow on each of it's k incoming arcs in order to send the proper amount
 247 of flow to B and C . Assume without loss of generality that the $k/2$ copies of sa are going
 248 to be used by $x_1, x_2, \dots, x_{k/2}$ and the $k/2$ copies of fa are left for $x_{k/2+1}, x_{k/2+2}, \dots, x_k$.
 249 Since the in-degree of f is also equal to k , f can receive at most λ units of flow of each
 250 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,
 251 \mathcal{N} does not admits k arc-disjoint s -branching flows. ◀

252 **3 Networks on (k, λ, s) -sufficient digraphs containing the flows**

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

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



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

262 **► Theorem 4.** *Let D be a (k, λ, s) -sufficient multi-branching. Then the network $\mathcal{N} = (D, u)$,
 263 with $u \equiv \lambda$, admits k arc-disjoint s -branching flows.*

264 The following result is useful for proving Theorem 4.

265 **► Lemma 5.** *Let D be a (k, λ, s) -sufficient digraph and let W be a nonempty subset of
 266 $V(D) \setminus \{s\}$ such that $N_D^+(V(D) \setminus W) = \{w\}$, $w \in W$. Then $D[W]$ is (k, λ, w) -sufficient.*

267 **Proof.** Since w is the unique vertex of W which has in-neighbours in $V(D) \setminus W$, then we
 268 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
 269 $d_{D[W]}^-(X) = d_D^-(X)$. ◀

270 **Proof of Theorem 4.** We are going to use induction on the height h of the multi-branching
 271 with root s D to construct k arc-disjoint s -branching flows f_1, f_2, \dots, f_k in \mathcal{N} .

272 The base case is $h = 1$. In this case D is a star and, since it is (k, λ, s) -sufficient, there
 273 are at least k arcs from s to every other vertex and we can use one arc for each flow f_j , for
 274 $j \in [k]$.

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

277 Let B_{r_i} be the subgraph of B_s that is a multi-branching with root r_i , for each $i \in [p]$.
 278 Observe that, by Lemma 5, B_{r_i} is (k, λ, r_i) -sufficient and has height $h_i \leq q - 1$, thus, by
 279 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
 280 $d^-(r_i) \geq k \lceil \frac{|B_{r_i}|}{\lambda} \rceil$ and then we can use $\lceil \frac{|B_{r_i}|}{\lambda} \rceil$ of these arcs to send the proper amount of
 281 flow from s to r_i on each x_j to complete the k s -branching flows. ◀

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

282 Since a multi-path is also a multi-branching, Theorem 4 generalizes the result of [7] for
283 multi-paths.

284 ► **Theorem 6.** *If D is a (k, λ, s) -sufficient digraph in \mathcal{D}_ℓ , then the network $\mathcal{N} = (D, u)$ with
285 $u \equiv \lambda$ admits k arc-disjoint s -branching flows.*

286 **Proof.** Let D be a digraph in \mathcal{D}_ℓ that is (k, λ, s) -sufficient. By definition, D is composed
287 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
288 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
289 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 $d_D^-(v_j^i) \geq k\lceil j/\lambda \rceil$,
290 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,
291 since the only arcs entering in it are those arriving at v_j^i . Considering only the vertices with
292 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

$$293 \quad r(P_i) = \begin{cases} k, & \text{if } p_i < \lambda, \\ \min\{r_{j\lambda}^i \mid 1 \leq j \leq \lfloor p_i/\lambda \rfloor\}, & \text{otherwise.} \end{cases}$$

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

296 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
297 $i \in [q]$. 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 $j_i \in [\lfloor p_i/\lambda \rfloor]$
298 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
299 the set $X = \bigcup_{i=1}^q X_i \cup \{t\}$. We have that

$$300 \quad d_D^-(X) = \sum_{i=1}^q d_D^-(X_i) + \sum_{i=q+1}^\ell e_i = k \sum_{i=1}^q j_i + \sum_{i=1}^q r(P_i) + \sum_{i=q+1}^\ell e_i = k \sum_{i=1}^q j_i + k'. \quad (2)$$

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

$$302 \quad d_D^-(X) \geq k \left\lceil \frac{|X|}{\lambda} \right\rceil = k \left\lceil \frac{1}{\lambda} + \frac{\lambda \sum_{i=1}^q j_i}{\lambda} \right\rceil = k \sum_{i=1}^q j_i + k. \quad (3)$$

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

305 ► **Claim 7.** There are k arc-disjoint flows x_1^i, \dots, x_k^i on the network $\mathcal{N}_i = (P_i, u)$, with
306 $u \equiv \lambda$, 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 flows
307 in $\mathcal{N}_i - t$.

308 **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
309 must receive $j + 1$ units of flow on the first m_i flows and j units on remaining $k - m_i$ flows.
310 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
311 of flow on each one of the k flows and we only have to argue how to send the extra unit
312 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
313 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
314 $r(P_i) = 0$ then $m_i = 0$ and no extra unit is needed). Otherwise, j is not multiple of λ and
315 when we send j units of flow through $\lceil j/\lambda \rceil$ arcs, there is an arc that will not be used in
316 the maximum capacity and so we can use it to send the extra unit. Applying this method
317 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
318 can send the extra unit of each flow to t because $m_i \leq e_i$. ◁

319 For each $i \in [\ell]$, we compute the flows $x_1^i, x_2^i, \dots, x_k^i$ as in the Claim 7. Observe that k' is
 320 the number of flows that reach t . If $k' = k$, then we can rename these k flows in such a way
 321 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
 322 reaches t and we modify it so that it doesn't reach t anymore, and we repeat this process
 323 until there are only k flows that reach t . ◀

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

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

332 ▶ **Theorem 8** (Max-flow min-cut). *In any network $\mathcal{N} = (D, u)$ with source s and sink t , the*
 333 *value of a maximum flow is equal to the capacity of a minimum cut.*

334 ▶ **Theorem 9.** [2] *Let k be an integer and $\mathcal{N} = (D, u)$ be a network with $u \equiv 1$ and a*
 335 *prescribed balance vector b such that $b \not\equiv 0$. There exist k arc-disjoint flows in \mathcal{N} , all with*
 336 *balance vector b , if and only if \mathcal{N} has a flow f with balance vector $b_f \equiv kb$. Hence one can*
 337 *decide the existence of these flows in polynomial time.*

338 ▶ **Lemma 10** (Adapted from [3]). *Given a network $\mathcal{N} = (D, u)$ and a prescribed balance*
 339 *vector b . Let $M = \sum_{v: b(v) > 0} b(v)$ and let $\mathcal{N}' = (D', u')$ be a network defined as follows:*

- 340 ■ $V(D') = V(D) \cup \{s', t'\};$
- 341 ■ $A(D') = A(D) \cup \{s'u : u \in V(D), b(u) > 0\} \cup \{vt' : v \in V(D), b(v) < 0\};$
- 342 ■ $u'(a) = u(a), \forall a \in A(D), u'(s'u) = b(u)$ if $b(u) > 0$ and $u'(vt') = -b(v)$ if $b(v) < 0$.

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

345 ▶ **Theorem 11.** *Let D be a (k, λ, s) -sufficient digraph. Then, if $\lambda = 1$ or $k = 1$, the network*
 346 *$\mathcal{N} = (D, u)$ with $u \equiv \lambda$ admits k arc-disjoint s -branching flows.*

347 **Proof.** We start by showing a result that is common to both cases. Let b_f be the following
 348 balance vector: $b_f(s) = k(n-1)$ and $b_f(v) = -k$, for all $v \in V(D) - s$, where $n = |V(D)|$.
 349 Let \mathcal{N}' be the network obtained from \mathcal{N} and b_f , as described in Lemma 10, and (S, \bar{S}) be
 350 a minimum (s', t') -cut in \mathcal{N}' . We show that $c(S, \bar{S}) = k(n-1)$ which, by the Theorem 8,
 351 implies that there is an (s', t') -flow with value $k(n-1)$ in \mathcal{N}' . Remark that, by construction
 352 of \mathcal{N}' , $u'(s's) = k(n-1)$, $u'(vt') = k$ and $u'(a) = \lambda$ for every $a \in A(D)$. It's easy to see that
 353 $c(S, \bar{S}) \leq k(n-1)$ because $c(s', V(D') - s') = k(n-1)$. Then, we might assume that $s \in S$
 354 and we define $X = \bar{S} \setminus \{t\}$. It follows that

$$355 \quad c(S, \bar{S}) = c(S, X) + c(S, \{t'\}) = \lambda d_D^-(X) + k(n - |X| - 1). \quad (4)$$

356 Case $\lambda = 1$. Since D is $(k, 1, s)$ -sufficient, $d_D^-(X) \geq k|X|$ and resuming (4), we have
 357 $c(S, \bar{S}) \geq k|X| + k(n - |X| - 1) = k(n-1)$. Thus, $c(S, \bar{S}) = k(n-1)$ and, by the Lemma 10,
 358 \mathcal{N} admits a flow with balance b_f . The result follows from Theorem 9 which states that \mathcal{N}
 359 admits k arc-disjoint s -branching flows if and only if it admits a single flow f with balance
 360 b_f .

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

361 Case $k = 1$. Here, D is $(1, \lambda, s)$ -sufficient, which means that $d_D^-(X) \geq (|X|/\lambda)$, and so
 362 $\lambda d_D^-(X) \geq |X|$. Replacing this in (4), we obtain $c(S, \bar{S}) \geq |X| + n - |X| - 1 = n - 1$. Thus,
 363 $c(S, \bar{S}) = k(n - 1)$. Similarly to the previous case, we conclude, by the Lemma 10, that \mathcal{N}
 364 admits a flow f with balance b_f . As $k = 1$, we have that f is a branching flow and the result
 365 follows. \blacktriangleleft

366 We also show that networks on $(2, n - 2, s)$ -sufficient digraphs contain 2 arc-disjoint
 367 s -branching flows. This is a slightly improvement on the result by [2] when $k = 2$, which
 368 states that Edmonds' characterizaion of digraphs containing 2 arc-disjoint s -branchings also
 369 characterizes the existence of 2 arc-disjoint s -branching flows. With our terminology, the
 370 result in [2] states the following:

371 **► Theorem 12.** [2] *If D is a $(k, n - 1, s)$ -sufficient digraph, then the network $\mathcal{N} = (D, u \equiv$
 372 $n - 1)$ admits k arc-disjoint s -branching flows.*

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

375 **Proof.** Applying Theorem 12 to the network $\mathcal{N}' = (D, u \equiv n - 1)$, we obtain two arc-disjoint
 376 s -branching flows f'_b and f'_r . By a result of [2], we may assume that the arcs used by f'_b and
 377 f'_r induce s -branchings in D . We colour blue the arcs used by f'_b , red the arcs used by f'_r ,
 378 and black the arcs not used by any of those flows. We remark that, since the blue arcs and
 379 the red arcs form a s -branching each, every vertex in $V(D) \setminus \{s\}$ is the head of exactly one
 380 blue and exactly one red arc.

381 If every $e \in A(D)$ has $f'_b(e) \leq n - 2$ and $f'_r(e) \leq n - 2$, then both flows can be transformed
 382 into arc disjoint s -branching flows f_b and f_r in \mathcal{N} by simply choosing $f_b(e) = f'_b(e)$ and
 383 $f_r(e) = f'_r(e)$ for all $e \in A(D)$. Assume now that this is not the case and let $X = V(D) \setminus \{s\}$.

384 If there are arcs e and a with $f'_b(e) = f'_r(a) = n - 1$ then, since $|X| = n - 1$ and D
 385 is $(k, n - 2, s)$ -sufficient, there are two black arcs e' and a' entering X and thus we can
 386 construct two arc-disjoint s -branching flows f_b and f_r in \mathcal{N} as follows: for every blue arc y
 387 in the unique blue path from s to the head of e' , define $f_b(y) = f'_b(y) - 1$, define $f'_b(e') = 1$,
 388 and construct f_r similarly using a' to route one unit of flow from s to X . Without loss of
 389 generality, we can now assume that there is a blue arc e with $f'_b(e) = n - 1$ from s to a vertex
 390 $v \in X$ and that $f'_r(a) \leq n - 2$ for every $a \in E(D)$. Notice that every arc entering X has tail
 391 s .

392 If there is a black arc entering X , then we can apply the same argument used in the
 393 previous case to construct an s -branching flow f_b in \mathcal{N} that is arc disjoint from f'_r . Thus we
 394 can assume that every arc from s to X is either blue or red and that, from such arcs, exactly
 395 one is blue and the remaining are red. Notice that there must be at least three red arcs from
 396 s to X since $|X| \geq n - 1$ and D is $(2, n - 2, s)$ -sufficient.

397 Since each vertex in X is incident to exactly one red arc, there is a vertex $w \in X$ such
 398 that the arc sw is red and there is no red path from w to the head v of e . Let R be the
 399 arborescence formed by the set of vertices reachable from w using only red arcs, together
 400 with such red arcs. Clearly there are no red arcs entering or leaving $V(R)$ since f'_r is an
 401 s -branching flow. Finally, let B be the set of blue arcs entering $V(R)$ together with all
 402 blue arcs between vertices in $V(R)$. We now construct two arc disjoint s arborescences by
 403 recolouring blue the arcs in $A(R)$ and red the arcs in B .

404 Let F_r be the arborescence on $V(D)$ containing only the set of red arcs, minus the arcs
 405 in $A(R)$, added of the arcs in B , and F_b be the arborescence on $V(D)$ containing only set of
 406 blue arcs, minus the arcs in B , added of the arcs in $A(R)$. Formally, $A(F_r) = (\{e \mid f'_r(e) >$

0} \setminus A(R)) \cup B and $A(F_B) = (\{e \mid f'_b(e) > 0\} \setminus B) \cup A(R)$. We claim that F_r and F_b are arc-disjoint s -arborescences. First, consider F_b . Since R is an arborescence and F_b preserves the blue arcs leaving $V(R)$, there is path in F_b from s to every other arc in $V(D)$. For F_r , first notice that there is an arc $tu \in B$ for every $u \in V(R)$ since every vertex in $V(D)$ is the head of one blue arc. Moreover, every vertex in $V(D) \setminus V(R)$ is reachable from s through a path using only red arcs in **not** in $A(R)$ and thus we can reach every vertex in $V(R)$ from s by using red arcs **not** in $A(R)$ and then arcs in B , and the claim follows. Finally, the fact that there are at least three red arcs entering arcs implies that F_r contains at least two arcs entering X and, since $v \notin V(R)$ and $sw \in F_b$, there are two arcs in F_b entering X . Thus we can construct two arc-disjoint s -branching flows in $(D, n - 2)$ by applying ideas similar to the previous cases where we assumed that there were two black arcs entering X , and the result follows. ◀

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

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

► **Theorem 14.** *Let $\mathcal{N} = (D, u)$ be a network with $u \equiv \lambda$ and $\lambda \geq 2$, and p be an integer 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 arc-disjoint s -branching flows.*

Proof. By Theorem 1, there are pk arc-disjoint s -branchings B_1, \dots, B_{pk} in D . For $i \in [k]$, 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 $\mathcal{N}_i = (D_i, u)$.

Now, for every non-empty $X \subseteq V(D) \setminus \{s\}$, we conclude that $d_{D_i}^-(X) \geq p$ since each B_i with $i \in [pk]$ must reach X at least once. Moreover, by our choice of p , we have that every \mathcal{N}_i is $(1, \lambda, s)$ -sufficient. Thus, by Theorem 11, we have that each \mathcal{N}_i admits an s -branching flow x_i . Finally, as every arc of D appears in at most one digraph D_i , we conclude that the s -branching flows x_1, \dots, x_k are pairwise arc-disjoint. The result follows since each of these flows can be promptly used to construct an s -branching flow in \mathcal{N} , by copying arcs and flow functions used by each flow x_j with $j \in [k]$. ◀

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

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

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

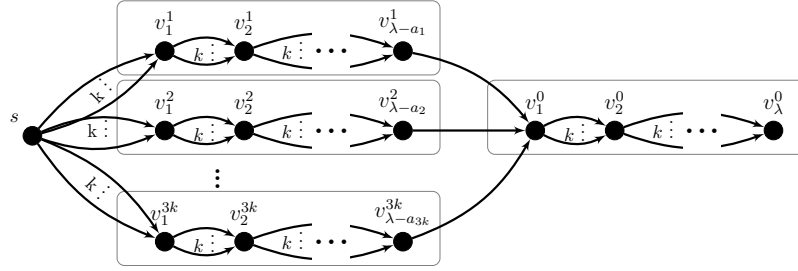
Proof. We are going to reduce the 3-PARTITION problem to our problem. Given a set of natural numbers $S = \{a_1, a_2, \dots, a_{3k}\}$ and a positive integer λ such that $\lambda/4 < a_i < \lambda/2$, for

23:12 A study of an Edmonds' like property to arc-disjoint branching flows

451 each $a_i \in S$, and such that $\sum_{i=1}^{3k} a_i = k\lambda$, the 3-PARTITION problem consists in deciding the
 452 following: can S be partitioned in k subsets S_1, S_2, \dots, S_k so that $\sum_{a_j \in S_i} a_j = \lambda$, $1 \leq i \leq k$?
 453 Observe that each S_i should be composed by exactly 3 elements of S .

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

457 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,
 458 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
 459 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 .
 460 See Figure 4. Finally, we set $u \equiv \lambda$ to conclude the construction.



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

461 First, let us check that D is (k, λ, s) -sufficient. By contradiction, assume that there
 462 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
 463 $|V - s| = (2k + 1)\lambda$ and $d_D^-(V - s) = k(3k)$. Recall that the in-degree of every vertex of P_i
 464 is k and thus the in-degree of X is at least k . Therefore X cannot be entirely contained
 465 in a unique P_i , $i \in [3k]$, once, in this case, $|X| \leq \lambda$ and then $\lceil |X|/\lambda \rceil = 1$. Similarly,
 466 X cannot be entirely contained in P_0 , since there are $3k$ arcs arriving in v_1^0 . Now, let
 467 $I = \{i : 1 \leq i \leq 3k, V(P_i) \cap X \neq \emptyset\}$. Observe that $d_D^-(X) \geq k|I|$. We are going now to
 468 consider the following cases:

- 469 (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
 470 contradiction to the choice of X .
 471 (ii) $X \cap V(P_0) \neq \emptyset$. Then $|X| \leq \lambda|I| + \lambda = \lambda(|I| + 1)$. Take the smallest j such that $v_j^0 \in X$.
 472 We analyze two possibilities:

473 (ii.1) $j > 1$. Then there are k arcs $v_{j-1}^0 v_j^0$ that enter X , that is, $d^-(X) \geq k|I| + k =$
 474 $k(|I| + 1) \geq k \lceil |X|/\lambda \rceil$ since I is a positive integer and $|X| \leq \lambda(|I| + 1)$ implying
 475 $(|I| + 1) \geq |X|/\lambda$, a contradiction.

476 (ii.2) $j = 1$. Therefore $v_1^0 \in X$ and $d^-(X) \geq k|I|$. Again, we consider some cases:
 477 (ii.2.1) If there are at least k different values of i for which $v_{\lambda-a_i}^i \notin X$, then
 478 such vertices contribute with k to the in-degree of X , and again we have
 479 $d^-(X) \geq k|I| + k$.

480 (ii.2.2) If there is only $1 \leq l < k$ different values of i for which $v_{\lambda-a_i}^i \notin X$,
 481 then, in order to complete the argument, we need to estimate more
 482 precisely the size of X . Recall that $\lambda/4 < a_i < \lambda/2$. So, $|X| \leq \lambda|I| -$
 483 $\frac{\lambda|I|}{4} + \lambda = \lambda(\frac{3|I|}{4} + 1)$. Observe that the amount of arcs entering X is

484 $k|I| + l = \frac{3k|I|}{4} + \frac{k|I|}{4} + l$. Since, in this case, $|I| > 2k$, we have that
 485 $\frac{k|I|}{4} \geq \frac{k^2}{2} > k$ if $k > 2$, which we can consider true by the definition of
 486 the 3-PARTITION problem ($k < 3$ would be directly a negative instance).
 487 Then, $d^-(X) \geq \frac{3k|I|}{4} + k = k(\lambda(\frac{3|I|}{4} + 1)/\lambda)$, a contradiction.

488 (ii.2.3) If all the last vertices are in X , then $d^-(X) \geq d^-(V - s)$, and we have
 489 another contradiction, since $|X| < |V(D) - s|$.

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

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

494 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
 495 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
 496 $\sum_{a_j \in S_i} a_j = \lambda$, the three arcs $v_{\lambda - a_j}^j$ for $a_j \in S_i$ will send the amount of flow needed to
 497 complete the branching flow in P_0 .

498 Finally, we assume that there exist k arc-disjoint s -branching flows x_1, x_2, \dots, x_k in
 499 \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
 500 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,
 501 as $|V(P_i)| = \lambda - a_i$, v_1^i must receive more than λ units of flow in x_t , that is, x_t uses at least
 502 two copies of the arc sv_1^i . This is a contradiction because there are only k copies of sv_1^i and
 503 each flow must use exactly one copy. Thus, the claim follows. Each x_r must reach v_1^0 with λ
 504 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
 505 the correspondent a_j forms a S_r . Then S_1, S_2, \dots, S_k is an *yes* instance of the 3-PARTITION
 506 problem. ◀

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

510 5 Concluding remarks

511 In this work, we studied the characterization of networks admitting k arc-disjoint s -branching
 512 flows. We showed that, in some cases (Theorems 4 and 6), an “Edmonds like” condition,
 513 which we call (k, λ, s) -sufficiency, is enough to guarantee the existence of the k arc-disjoint
 514 s -branching flows but, in other cases, a stronger statement is needed (Section 2). This
 515 proves that (k, λ, s) -sufficiency in general is not enough to guarantee the existence of the
 516 desired s -branching flows in networks, although the cases for which it does are still interesting
 517 because, among other reasons, they have been resulting in polynomial-time algorithms to find
 518 the flows. Some of the positive results we prove consider restrictions on the structure of the
 519 digraphs on which the networks are built: we show that the analyzed property guarantees the
 520 existence of the flows on multi-spindles and on multi-branchings. In other words, we prove
 521 that the flows always exist in (k, λ, s) -sufficient networks built on digraphs whose underlying
 522 simple graphs are trees or spindles.

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

529 — References

- 530 1 R. Ahuja. *Network flows : theory, algorithms, and applications*. Prentice Hall, Englewood
531 Cliffs, N.J, 1993.
- 532 2 J. Bang-Jensen and S. Bessy. (Arc-)disjoint flows in networks. *Theoretical Computer Science*,
533 526:28–40, 2014. doi:<https://doi.org/10.1016/j.tcs.2014.01.011>.
- 534 3 J. Bang-Jensen and G. Z. Gutin. *Digraphs: theory, algorithms and applications*. Springer
535 Science & Business Media, 2008.
- 536 4 J. Bang-Jensen, F. Havet, and A. Yeo. The complexity of finding arc-disjoint branching flows.
537 *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)
538 10.012.
- 539 5 S. Bessy, F. Hörsch, A.K. Maia, D. Rautenbach, and I. Sau. FPT algorithms for packing
540 k -safe spanning rooted sub(di)graphs, 2021. arXiv:2105.01582.
- 541 6 J. A. Bondy and U. S. R. Murty. *Graph theory*, volume 244 of *Graduate Texts in Mathematics*.
542 Springer, New York, 2008. URL: <http://dx.doi.org/10.1007/978-1-84628-970-5>, doi:10.1007/
543 978-1-84628-970-5.
- 544 7 J. Costa, C. Linhares Sales, R. Lopes, and A.K. Maia. Um estudo de redes com fluxos
545 ramificados arco-disjuntos. *Matemática Contemporânea*, 46:230–238, 2019.
- 546 8 J. Edmonds. Edge-disjoint branchings. *Combinatorial Algorithms*, 1973.
- 547 9 S. Even, A. Itai, and A. Shamir. On the complexity of timetable and multicommodity flow
548 problems. *SIAM Journal on Computing*, 5(4):691–703, 1976. doi:10.1137/0205048.
- 549 10 D. R. Ford and D. R. Fulkerson. *Flows in Networks*. Princeton University Press, Princeton,
550 NJ, USA, 1962.
- 551 11 L. R. Ford and D. R. Fulkerson. Maximal flow through a network. *Canadian Journal of*
552 *Mathematics*, 8:399–404, 1956. doi:10.4153/CJM-1956-045-5.
- 553 12 A. Frank. Covering branchings. *Acta Scientiarum Mathematicarum (Szeged)*, 41:77–81, 1979.
- 554 13 R. Impagliazzo, R. Paturi, and F. Zane. Which problems have strongly exponential complexity?
555 *Journal of Computer and System Sciences*, 63(4):512–530, 2001. doi:[https://doi.org/10.](https://doi.org/10.1006/jcss.2001.1774)
556 1006/jcss.2001.1774.
- 557 14 L. Lovász. On two minimax theorems in graph. *Journal of Combinatorial Theory,*
558 *Series B*, 21(2):96 – 103, 1976. URL: [http://www.sciencedirect.com/science/article/pii/](http://www.sciencedirect.com/science/article/pii/0095895676900496)
559 [0095895676900496](http://www.sciencedirect.com/science/article/pii/0095895676900496), doi:[https://doi.org/10.1016/0095-8956\(76\)90049-6](https://doi.org/10.1016/0095-8956(76)90049-6).
- 560 15 K. Menger. Zur allgemeinen kurventheorie. *Fundamenta Mathematicae*, 10(1):96–115, 1927.
561 URL: <http://eudml.org/doc/211191>.
- 562 16 A. Schrijver. *Combinatorial optimization: polyhedra and efficiency*, volume 24. Springer
563 Science & Business Media, 2003.
- 564 17 Y. Shiloach. Edge-disjoint branching in directed multigraphs. *Inf. Process. Lett.*, 8(1):24–27,
565 1979. doi:10.1016/0020-0190(79)90086-3.
- 566 18 R. Tarjan. A good algorithm for edge-disjoint branching. *Information Processing Letters*,
567 3(2):51–53, 1974. doi:10.1016/0020-0190(74)90024-6.
- 568 19 A. Marino V. Campos, R. Lopes and A. Silva. Edge-disjoint branchings in temporal graphs.
569 In *Proc. of the 31st International Workshop on Combinatorial Algorithms (IWOCA)*, volume
570 12126 of *LNCS*, pages 112–115, 2020. doi:10.1007/978-3-030-48966-3_9.