Discrete Mathematics and Theoretical Computer Science ....
.... **vol. 25:3 special issue ICGT'22 #2 (2023)**

$(k-2)$-*linear connected components in hypergraphs of rank* k

Florian Galliot[1*] Sylvain Gravier[1] Isabelle Sivignon[2]

1. *Univ. Grenoble Alpes, CNRS, Institut Fourier, Grenoble, France*
2. *Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-lab, Grenoble, France*

revisions 25th Oct. 2022, 21st July 2023; accepted 19th Oct. 2023.

We define a *q-linear path* in a hypergraph $\mathcal{H}$ as a sequence $(e_1, \dots, e_L)$ of edges of $\mathcal{H}$ such that $|e_i \cap e_{i+1}| \in [\![1, q]\!]$ and $e_i \cap e_j = \varnothing$ if $|i-j| > 1$. In this paper, we study the connected components associated to these paths when $q = k-2$ where k is the rank of $\mathcal{H}$. If $k = 3$ then $q = 1$ which coincides with the well-known notion of *linear path* or *loose path*. We describe the structure of the connected components, using an algorithmic proof which shows that the connected components can be computed in polynomial time. We then mention two consequences of our algorithmic result. The first one is that deciding the winner of the Maker-Breaker game on a hypergraph of rank 3 can be done in polynomial time. The second one is that tractable cases for the NP-complete problem of "Paths Avoiding Forbidden Pairs" in a graph can be deduced from the recognition of a special type of line graph of a hypergraph.

Keywords: hypergraph, 3-uniform, path, linear path, chain, connectivity, polynomial-time algorithm

1 Introduction

There are many possible definitions for a path between two vertices in a hypergraph. Each one has its own associated connectivity problem, consisting in the algorithmic computation of the connected components and the potential study of their structure. Possible fields where such problems apply include system security [Guzzo et al. (2014)] on undirected hypergraphs as well as propositional logic [Gallo et al. (1993)], system transfer protocols [Thakur and Tripathi (2009)] or computational tropical geometry [Allamigeon (2014)] on directed hypergraphs.

In an undirected hypergraph, a *linear path* (or *loose path*) is a sequence of edges such that any two consecutive edges intersect on exactly one vertex and any two non-consecutive edges do not intersect. Our main motivation is the connectivity problem associated with linear paths in 3-uniform hypergraphs. The existence of such paths is the subject of numerous extremal results [Omidi and Shahsiah (2014)] [Jackowska (2015)] [Jackowska et al. (2016)] [Wu and Peng (2021)]. For instance, [Jackowska et al. (2016)] determines the Turán number of the 3-uniform linear path

*Affiliation at the time of publication: Univ. Bordeaux, CNRS, Bordeaux INP, LaBRI, Talence, France.

ISSN 1365–8050

of length 3, so that a 3-uniform hypergraph on $n \geq 8$ vertices with at least $\binom{n-1}{2}$ edges necessarily contains a 3-uniform linear path of length 3. Such results are proven using counting methods. The study of linear structures in potentially sparser hypergraphs, however, requires tools of a qualitative nature. It then seems reasonable to start by studying the linear connected components. In order to describe their structure, we develop methods that actually generalize to hypergraphs of rank $k \geq 4$ when replacing linearity with a notion of $(k-2)$-*linearity*.

We thus introduce the general concept of q-linear path, where any two consecutive edges intersect on between 1 and q vertices (and non-consecutive edges do not intersect). Extremal results also exist on paths with similar restrictions on the size of the intersections, for example paths where any two consecutive edges must intersect on exactly q vertices [Tomescu (2012)] [Dudek et al. (2017)] with emphasis on the linear case $q = 1$ [Füredi et al. (2014)] [Gu et al. (2020)]. Throughout this article, let $\mathcal{H}$ be a hypergraph of rank k: as for any hypergraph, we denote its vertex set by $V(\mathcal{H})$ and its edge set by $E(\mathcal{H})$. Define the q-linear connected component of $x^* \in V(\mathcal{H})$ as the set $LCC^q_{\mathcal{H}}(x^*)$ of all vertices x such that there exists a q-linear path between x^* and x in $\mathcal{H}$. We will see that q-linear paths do not define a transitive relation, so that the q-linear connected components of $\mathcal{H}$ do not form a partition of $V(\mathcal{H})$, unlike most other connectivity problems. This paper is a study of the q-linear connected components of $\mathcal{H}$ in the case $q = k-2$, meaning that we only prohibit *tight* intersections of size $k-1$. Linear paths in 3-uniform hypergraphs correspond to the case $k = 3$ *i.e.* $q = 1$. Our first main result describes the structure of the subhypergraph $\mathcal{H}[LCC^{k-2}_{\mathcal{H}}(x^*)]$ induced by a $(k-2)$-linear connected component.

The proof of the structural result is algorithmic and provides us with a way to compute the $(k-2)$-linear connected components in polynomial time. More precisely, our second main result is an algorithm that computes $LCC^{k-2}_{\mathcal{H}}(x^*)$ in $O(m^2k)$ time where $m = |E(\mathcal{H})|$, which remains polynomial even if k is part of the input. This result has consequences on two algorithmic problems that have long existed in the literature.

The first one is the problem of deciding the winner of the *Maker-Breaker* positional game. Two players, Maker and Breaker, take turns picking vertices of a hypergraph $\mathcal{H}$: Maker wins if she owns all the vertices of some edge of $\mathcal{H}$, and Breaker wins if he prevents this from happening. The problem of deciding the winner of the game with optimal play is trivially tractable for hypergraphs of rank 2, and is known to be PSPACE-complete for 6-uniform hypergraphs [Rahman and Watson (2021)]. In a separate paper [Galliot et al. (2022)], we show tractability for hypergraphs of rank 3, by reducing to the linear path existence problem in 3-uniform hypergraphs and using the polynomial-time algorithm provided by the present paper. This validates part of a conjecture by [Rahman and Watson (2020)].

The second one is the "Paths Avoiding Forbidden Pairs" problem (known as PAFP) which, given two vertices x, y in a graph G with blue and red edges, asks whether there exists a blue induced path between x and y in G. Indeed, consider a bicolored version of the line graph of a hypergraph, where a blue (resp. red) edge indicates an intersection of size between 1 and $k-2$ (resp. of size $k-1$): if G is the bicolored line graph of some k-uniform hypergraph $\mathcal{H}$, then there exists a blue induced path between two vertices of G if and only if there exists a $(k-2)$-linear path in $\mathcal{H}$ between the corresponding (hyper)edges. Since our connectivity problem is solvable in polynomial time, the study of the bicolored line graph recognition problem has the potential to unearth new tractable cases for PAFP, which is known to be NP-complete in general [Gabow et al. (1976)].

After some basic definitions given in Section 2, including the introduction of q-linear paths, Section 3 presents structures that are specific to the case $q = k-2$ as well as some of their properties. It is then shown algorithmically in Section 4 that these structures describe the $(k-2)$-linear connected components, which can be computed in polynomial time: these are our two main results. Finally, Section 5 addresses the links that our algorithmic problem has with the Maker-Breaker game and the PAFP problem. We end by formulating some open problems that arise from our study.

2 q-linear paths

2.1 Sequences of edges

Definition 2.1. A *sequence of edges of* $\mathcal{H}$ is some $\overrightarrow{P} = (e_1, \ldots, e_L)$ where $e_i \in E(\mathcal{H})$ for all $1 \leq i \leq L$. The case $L = 0$ is authorized: we may then denote $\overrightarrow{P} = ()$.

Notation 2.2. Let $\overrightarrow{P} = (e_1, \ldots, e_L)$ be a sequence of edges of $\mathcal{H}$.

- We define $V(\overrightarrow{P}) := e_1 \cup \ldots \cup e_L \subseteq V(\mathcal{H})$ and $E(\overrightarrow{P}) := \{e_1, \ldots, e_L\} \subseteq E(\mathcal{H})$.
- Let $\overrightarrow{Q} = (e'_1, \ldots, e'_M)$ be another sequence of edges of $\mathcal{H}$. We denote by $\overrightarrow{P} \oplus \overrightarrow{Q}$ the concatenation of $\overrightarrow{P}$ and $\overrightarrow{Q}$, that is $\overrightarrow{P} \oplus \overrightarrow{Q} := (e_1, \ldots, e_L, e'_1, \ldots, e'_M)$.

2.2 Description of the problem

Definition 2.3. A *path* in $\mathcal{H}$ is a sequence $\overrightarrow{P} = (e_1, \ldots, e_L)$ of edges of $\mathcal{H}$ such that one can write $V(\overrightarrow{P}) = \{x_1, \ldots, x_N\}$ and $e_i = \{x_{s_i}, x_{s_i+1}, \ldots, x_{f_i}\}$ with $s_i < s_{i+1} \leq f_i < f_{i+1}$ for all $1 \leq i \leq L-1$. Note that $e_i \cap e_{i+1} \neq \varnothing$ for all $1 \leq i \leq L-1$. The path is deemed *simple* if $e_i \cap e_j = \varnothing$ for all $1 \leq i,j \leq L$ such that $|i-j| > 1$. See Figure 1.

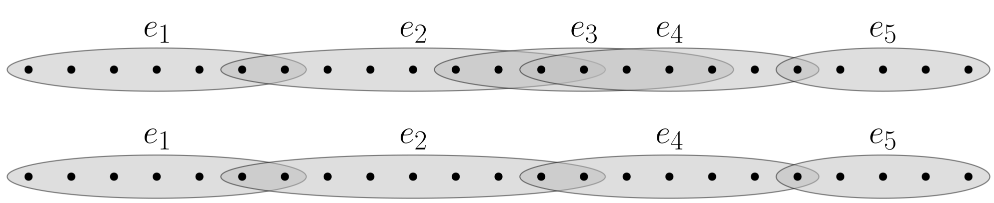

Fig. 1: The top path is not simple because $e_2 \cap e_4 \neq \varnothing$. Removing e_3 yields a simple path (bottom).

We study paths with the additional *q-linearity property* that $|e_i \cap e_{i+1}| \leq q$ for some fixed integer q. Since we are only interested in existence questions, we can focus on simple such paths: indeed, from any path it is possible to extract a simple path by removing some edges if necessary, and this obviously preserves the q-linearity property. An equivalent definition is the following:

Definition 2.4. Let $q \geq 1$. A *q-linear path* in $\mathcal{H}$ is a sequence $\overrightarrow{P} = (e_1, \ldots, e_L)$ of edges of $\mathcal{H}$ such that for all $1 \leq i < j \leq L$: $|e_i \cap e_j| \begin{cases} \in [\![1, q]\!] & \text{if } j = i+1. \\ = 0 & \text{otherwise.} \end{cases}$.

Definition 2.5. Let $q \geq 1$ be an integer and let $X, Y \subseteq V(\mathcal{H})$ be nonempty such that $|X \cap Y| \leq q$. A *q-linear path from X to Y* in $\mathcal{H}$ is a q-linear path $\overrightarrow{P} = (e_1, \ldots, e_L)$ in $\mathcal{H}$ such that:
- If $X \cap Y \neq \varnothing$, then $L = 0$.
- If $X \cap Y = \varnothing$, then $L \geq 1$ and:
  (i) $X \cap e_1 \neq \varnothing$, and if $L \geq 2$ then $X \cap e_i = \varnothing$ for all $2 \leq i \leq L$.
  (ii) $Y \cap e_L \neq \varnothing$, and if $L \geq 2$ then $Y \cap e_i = \varnothing$ for all $1 \leq i \leq L-1$.

Whenever $X = \{x\}$, we may use the abuse of notation $X = x$ (same for Y). See Figure 2.

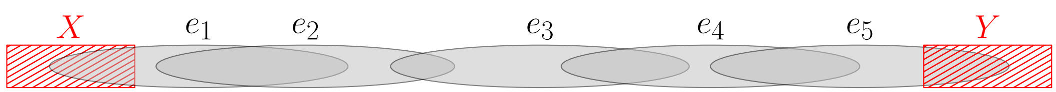

Fig. 2: Schematic representation of a q-linear path from X to Y.

Lemma 2.6. *Let $\overrightarrow{P} = (e_1, \ldots, e_L)$ be a q-linear path in $\mathcal{H}$ such that $L \geq 1$. Let $X, Y \subseteq V(\mathcal{H})$ be disjoint such that $X \cap e_1 \neq \varnothing$ and $Y \cap e_L \neq \varnothing$. Then $\overrightarrow{P}$ contains a q-linear path $\overrightarrow{Q}$ from X to Y in $\mathcal{H}$. More precisely: $\overrightarrow{Q} = (e_r, \ldots, e_s)$ where $s := \inf\{1 \leq i \leq L$ such that $e_i \cap Y \neq \varnothing\}$ and $r := \sup\{1 \leq i \leq s$ such that $e_i \cap X \neq \varnothing\}$.*

Proof: This is clear by minimality (resp. maximality) of s (resp. r). $\square$

Definition 2.7. Let $x \in V(\mathcal{H})$. The *q-linear connected component of x* in $\mathcal{H}$ is defined as:

$$LCC^q_{\mathcal{H}}(x) := \{y \in V(\mathcal{H}) \text{ such that there exists a } q\text{-linear path from } x \text{ to } y \text{ in } \mathcal{H}\}.$$

It is important to note that q-linear paths do not define a transitive relation, so that the q-linear connected components of a hypergraph do not necessarily form a partition of its vertex set. Indeed, the union of a q-linear path from x to y and a q-linear path from y to z does not necessarily contain a q-linear path from x to z. An illustration in the case $q = 1$ is provided in Figure 3 (this graphical representation of 3-uniform hypergraphs will be used throughout, with each edge pictured as a "claw" joining its three vertices). Therefore, the problem consisting in computing the q-linear connected component of a given vertex is nontrivial.

This problem reduces polynomially to the case where $\mathcal{H}$ is uniform. Indeed, if $\mathcal{H}$ is of rank k then let $\mathcal{H}_0$ be the k-uniform hypergraph obtained from $\mathcal{H}$ by adding $k - |e|$ new vertices to each edge e: it is easy to see that there exists a q-linear path from x to y in $\mathcal{H}$ if and only if there

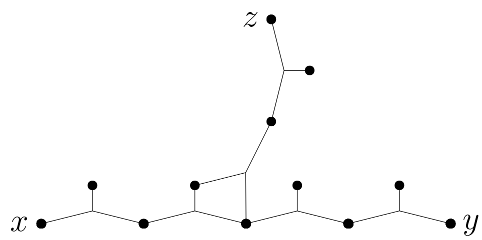

Fig. 3: There is no 1-linear path from x to z.

exists one in $\mathcal{H}_0$. We thus introduce the following decision problem:

HypConnectivity$_{k,q}$
Input : a k-uniform hypergraph $\mathcal{H}$ and two distinct vertices x, y of $\mathcal{H}$.
Output : YES if and only if there exists a q-linear path from x to y in $\mathcal{H}$.

The case $q = k-1$ corresponds to standard (*i.e.* non-constrained) connectivity in hypergraphs, which is tractable via a simple DFS/BFS-type search. We now address the case $q = k-2$.

3 $(k-2)$-linear paths in k-uniform hypergraphs

In this section, we suppose $\mathcal{H}$ is k-uniform with $k \geq 3$.

3.1 Extendable paths and islands

3.1.1 Principle

Let $x^* \in V(\mathcal{H})$ be the vertex whose $(k-2)$-linear connected component we wish to compute. The idea is to design an algorithm that searches through $E(\mathcal{H})$ and accepts edges under some guarantee that all their vertices are in $LCC_{\mathcal{H}}^{k-2}(x^*)$.

Consider the situation in the middle of the execution of the algorithm. Some edges have already been accepted, forming a subhypergraph $\mathcal{I}_1$ of $\mathcal{H}$ containing x^* such that: for all $x \in V(\mathcal{I}_1)$, there exists a $(k-2)$-linear path from x^* to x in $\mathcal{I}_1$. Now, the algorithm encounters some edge e intersecting both $V(\mathcal{I}_1)$ and $V(\mathcal{H}) \setminus V(\mathcal{I}_1)$, and needs to decide whether or not e should be accepted right away: let $x \in e \setminus V(\mathcal{I}_1)$, can we find a $(k-2)$-linear path from x^* to x made of edges in $E(\mathcal{I}_1) \cup \{e\}$?

The only way would be to use a $(k-2)$-linear path $\overrightarrow{P} = (e_1, \ldots, e_L)$ from x^* to $X := e \cap V(\mathcal{I}_1)$ in $\mathcal{I}_1$ (Lemma 2.6 ensures there exists one), and prolong it with the edge e to reach $e \setminus V(\mathcal{I}_1)$. However, though $\overrightarrow{P} \oplus (e) = (e_1, \ldots, e_L, e)$ is obviously $(k-2)$-linear if $|X| \leq k-2$, it might not be if $|X| = k-1$: indeed, in that case, if $X \subset e_L$ then $|e_L \cap e| = k-1$. On this account, if $|X| = k-1$ then we need $\overrightarrow{P}$ to not just be any $(k-2)$-linear path from x^* to X but to be one that satisfies $X \not\subset e_L$: such a path will be deemed (x^*, X)-*extendable*, because it can be prolonged by an edge that contains X while preserving the $(k-2)$-linearity. An illustration is given in Figure 4.

So, what property must $\mathcal{I}_1$ have if we want to be able to accept any edge intersecting both $V(\mathcal{I}_1)$ and $V(\mathcal{H}) \setminus V(\mathcal{I}_1)$? As we have just seen, the existence of a $(k-2)$-linear path from x^* to x in $\mathcal{I}_1$ for all $x \in V(\mathcal{I}_1)$ is not sufficient. Additionally to this, we would need the existence of an

(x^*, X)-extendable path in $\mathcal{I}_1$ for all $X \subset V(\mathcal{I}_1)$ of size $k-1$. If $\mathcal{I}_1$ satisfies these two properties, we will say $\mathcal{I}_1$ is an *island with entry* $\{x^*\}$.

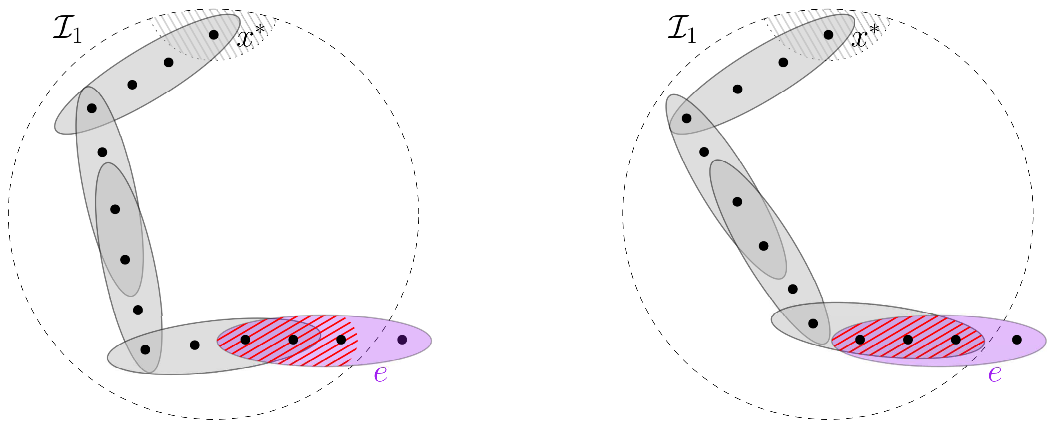

Fig. 4: Here $k = 4$ and $|X| = 3$ (the red hatched area is X). The grey path from x^* to X on the left is (x^*, X)-extendable, but the one on the right is not because its final edge contains X entirely.

However, the accepted edges might not always form an island. Suppose $\mathcal{I}_1$ is an island and we next discover an edge e_0 such that $|e_0 \cap V(\mathcal{I}_1)| = 1$ (so we accept e_0) *i.e.* e_0 is of the form $e_0 = \{x_1\} \cup \varepsilon$ where $e_0 \cap V(\mathcal{I}_1) = \{x_1\}$ and $|\varepsilon| = k-1$. Then the accepted edges do not form an island anymore: the only known $(k-2)$-linear paths from x^* to ε use e_0 so they contain ε entirely, meaning they are not (x^*, ε)-extendable. Suppose the next few accepted edges form a subhypergraph $\mathcal{I}_2$ that contains ε but is disjoint from $\mathcal{I}_1$, such that for all $x \in V(\mathcal{I}_2)$ there exists a $(k-2)$-linear path from ε to x in $\mathcal{I}_2$. The algorithm now encounters some edge e whose known vertices are in $\mathcal{I}_2$ (see Figure 5): should we accept e? Let $X := e \cap V(\mathcal{I}_2)$ and $y \in e \setminus X$. The only way to reach y from x^* is via $\overrightarrow{R} := \overrightarrow{P} \oplus (e_0) \oplus \overrightarrow{Q} \oplus (e)$ where $\overrightarrow{P}$ is a $(k-2)$-linear path from x^* to x_1 in $\mathcal{I}_1$ and $\overrightarrow{Q}$ is a $(k-2)$-linear path from ε to X in $\mathcal{I}_2$. We know such a $\overrightarrow{P}$ exists, however there are conditions on $\overrightarrow{Q} = (e_1, \ldots, e_L)$ for $\overrightarrow{R}$ to be $(k-2)$-linear:

- As before, if $|X| = k-1$ then we need $X \not\subset e_L$.
- Since $\varepsilon \subset e_0$, we also need $\varepsilon \not\subset e_1$.

Such a path $\overrightarrow{Q}$ will be deemed (ε, X)-*extendable* (this time, there are conditions at both ends of the path). In conclusion, to be able to accept any such e, we would need the existence of an (ε, X)-extendable path in $\mathcal{I}_2$ for all $X \subset V(\mathcal{I}_2)$ of size at most $k-1$. If $\mathcal{I}_2$ satisfies these two properties, we will say $\mathcal{I}_2$ is an *island with entry* ε.

We see the premises of the *archipelago* structure of $\mathcal{H}[LCC^{k-2}_{\mathcal{H}}(x^*)]$, which we are going to establish.

3.1.2 Definitions

We now give the formal definitions that we are going to use.

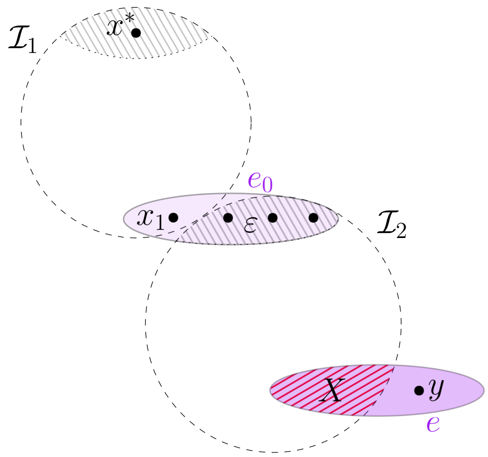

Fig. 5: Here $k = 4$ so $|\varepsilon| = 3$.

Definition 3.1. Let $X, Y \subseteq V(\mathcal{H})$ such that $1 \leq |X|, |Y| \leq k-1$ and $|X \cap Y| \leq k-2$. An (X,Y)-*extendable path* in $\mathcal{H}$ is a $(k-2)$-linear path $\overrightarrow{P} = (e_1, \dots, e_L)$ from X to Y in $\mathcal{H}$ with the additional property if $L \geq 1$ that $|e_1 \cap X| \leq k-2$ and $|e_L \cap Y| \leq k-2$.

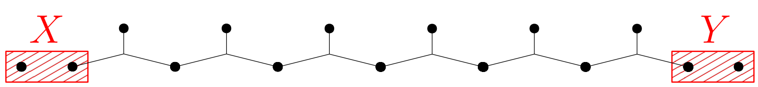

Fig. 6: An (X,Y)-extendable path in the case $k = 3$: the path contains exactly one vertex of X and one vertex of Y.

Note that the condition on X is empty if $|X| \leq k-2$: it is only when $|X| = k-1$ that we need to make sure that prolonging $\overrightarrow{P}$ with an edge containing X maintains the $(k-2)$-linearity (same for Y). Therefore, if $|X|, |Y| \leq k-2$, then an (X,Y)-extendable path is simply a $(k-2)$-linear path from X to Y. It is also important to keep in mind that the definition is dependent on X and Y: we do not define an "extendable path", we define an "(X,Y)-extendable path".

Definition 3.2. Let $\mathcal{I}$ be a subhypergraph of $\mathcal{H}$ and $\varepsilon \subset V(\mathcal{I})$ such that $1 \leq |\varepsilon| \leq k-1$. We say $\mathcal{I}$ is an *island with entry* ε if, for all $X \subset V(\mathcal{I})$ satisfying $1 \leq |X| \leq k-1$ (and $X \neq \varepsilon$ if $|\varepsilon| = k-1$), there exists an (ε, X)-extendable path in $\mathcal{I}$.

Example. The *empty island with entry* $\varepsilon \subset V(\mathcal{H})$, where $1 \leq |\varepsilon| \leq k-1$, is the island $\mathcal{I}$ with entry ε defined by $V(\mathcal{I}) = \varepsilon$ and $E(\mathcal{I}) = \varnothing$. It is an island because, for all $X \subset V(\mathcal{I})$ satisfying $1 \leq |X| \leq k-1$ (and $X \neq \varepsilon$ if $|\varepsilon| = k-1$), $\overrightarrow{P} = ()$ is an (ε, X)-extendable path in $\mathcal{I}$. This example is illustrated at the far left of Figure 7.

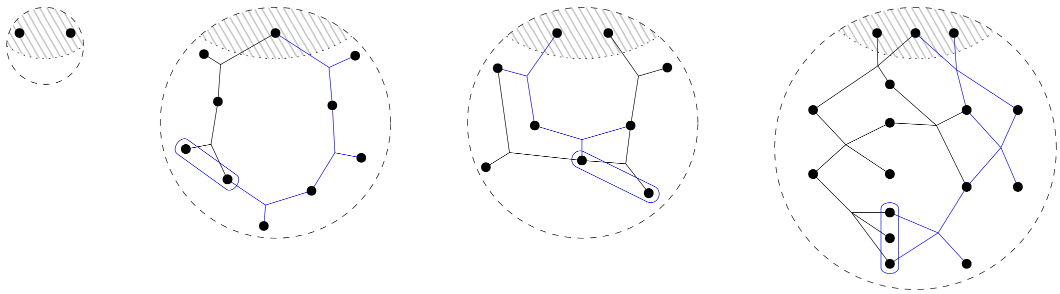

Fig. 7: Some islands for $k = 3$, except the far right one where $k = 4$ (with the same "claw" representation for edges). The grey hatched area will always represent the entry. For three of them, we show an (ε, X)-extendable path (in blue) for some X of size $k-1$ (circled in blue).

3.1.3 Extension lemmas

The notion of (X, Y)-extendable path has been introduced to prolong and compose $(k-2)$-linear paths. In that direction, we now prove two useful lemmas which are illustrated in Figures 8 and 9.

Lemma 3.3. *Let $A, B \subseteq V(\mathcal{H})$ such that $1 \leq |A|, |B| \leq k-1$ and $|A \cap B| \leq k-2$, and let $\overrightarrow{P}$ be an (A, B)-extendable path.*

- *If $B' \supset B$ is such that $|B'| \leq k-1$ and $B' \cap (A \cup V(\overrightarrow{P}) \cup B) = B$, then $\overrightarrow{P}$ is an (A, B')-extendable path.*
- *If $A' \supset A$ is such that $|A'| \leq k-1$ and $A' \cap (A \cup V(\overrightarrow{P}) \cup B) = A$, then $\overrightarrow{P}$ is an (A', B)-extendable path.*

Proof: By symmetry, we only need to prove the first assertion. First notice that $A \cap B = A \cap B'$, so that $|A \cap B'| \leq k-2$ as required in Definition 2.5.

- If $A \cap B' \neq \varnothing$ then $A \cap B \neq \varnothing$, hence $\overrightarrow{P} = ()$ which is an (A, B')-extendable path.
- If $A \cap B' = \varnothing$ then $A \cap B = \varnothing$, so we can write $\overrightarrow{P} = (e_1, \ldots, e_L)$ where $L \geq 1$. We already know $\overrightarrow{P}$ is $(k-2)$-linear, moreover the assumption on B' ensures that $\overrightarrow{P}$ is from A to B'. Finally, since $\overrightarrow{P}$ is (A, B)-extendable and $e_L \cap B' = e_L \cap B$, we have $|e_1 \cap A| \leq k-2$ and $|e_L \cap B'| = |e_L \cap B| \leq k-2$, therefore $\overrightarrow{P}$ is (A, B')-extendable. □

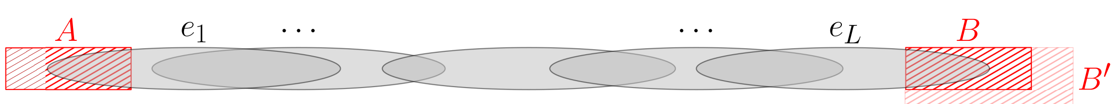

Fig. 8: Illustration of Lemma 3.3.

Lemma 3.4. *Let $A, B \subseteq V(\mathcal{H})$ such that $1 \le |A|, |B| \le k-1$ and $|A \cap B| \le k-2$, and let $\overrightarrow{P}$ be an (A,B)-extendable path. Let $C, D \subseteq V(\mathcal{H})$ such that $1 \le |C|, |D| \le k-1$ and $|C \cap D| \le k-2$, and let $\overrightarrow{Q}$ be a (C,D)-extendable path. We assume that $A \cup V(\overrightarrow{P}) \cup B$ and $C \cup V(\overrightarrow{Q}) \cup D$ are disjoint. If $e \in E(\mathcal{H})$ satisfies $e \cap (A \cup V(\overrightarrow{P}) \cup B) = B$ and $e \cap (C \cup V(\overrightarrow{Q}) \cup D) = C$, then $\overrightarrow{P} \oplus (e) \oplus \overrightarrow{Q}$ is an (A,D)-extendable path.*

Proof: Write $\overrightarrow{P} = (e_1, \dots, e_L)$ and $\overrightarrow{Q} = (e'_1, \dots, e'_M)$, and define $\overrightarrow{R} := \overrightarrow{P} \oplus (e) \oplus \overrightarrow{Q}$. Let us first check that $\overrightarrow{R}$ is a $(k-2)$-linear path. Any intersection between two edges of $\overrightarrow{R}$ is of one of four forms:

(1) $e_i \cap e_j$ or $e'_i \cap e'_j$.
Those are covered by the $(k-2)$-linearity of $\overrightarrow{P}$ and $\overrightarrow{Q}$ respectively.

(2) $e_i \cap e'_j$.
Those are empty because $V(\overrightarrow{P})$ and $V(\overrightarrow{Q})$ are disjoint by assumption.

(3) $e_i \cap e$ where $1 \le i \le L-1$ or $e \cap e'_i$ where $2 \le i \le M$.
By symmetry, we only address $e_i \cap e$. Since $\overrightarrow{P}$ is from A to B, we know $e_i \cap B = \varnothing$. Moreover $e \cap V(\overrightarrow{P}) \subseteq B$ by assumption, so $e_i \cap e = \varnothing$.

(4) $e_L \cap e$ or $e \cap e'_1$.
By symmetry, we only address $e_L \cap e$. Since $\overrightarrow{P}$ is (A,B)-extendable, we know $|e_L \cap B| \le k-2$, moreover the assumption on e implies $e_L \cap e = e_L \cap B$ hence $|e_L \cap e| \le k-2$.

We now verify that $\overrightarrow{R}$ is from A to D and is (A,D)-extendable. By symmetry, we only show the conditions on A, for which we distinguish two cases:

- If $L = 0$, then the first edge of $\overrightarrow{R}$ is e. We have $A \cap e = A \cap B$ by the assumption on e, where $A \cap B \ne \varnothing$ (because $L = 0$) and $|A \cap B| \le k-2$ (by assumption), therefore $1 \le |A \cap e| \le k-2$. It remains to show that $A \cap e'_i = \varnothing$ for all $1 \le i \le M$, which is obvious since A is disjoint from $V(\overrightarrow{Q})$.
- If $L \ge 1$, then the first edge of $\overrightarrow{R}$ is e_1. Since $\overrightarrow{P}$ is from A to B, we have $A \cap e_1 \ne \varnothing$ and $A \cap e_i = \varnothing$ for all $2 \le i \le L$. Moreover $|A \cap e_1| \le k-2$ because $\overrightarrow{P}$ is (A,B)-extendable. It remains to show that $A \cap e = \varnothing$, which is clear since $A \cap e \subseteq B$ and $A \cap B = \varnothing$ ($L \ge 1$), and that $A \cap e'_i = \varnothing$ for all $1 \le i \le M$, which is obvious since A is disjoint from $V(\overrightarrow{Q})$. $\square$

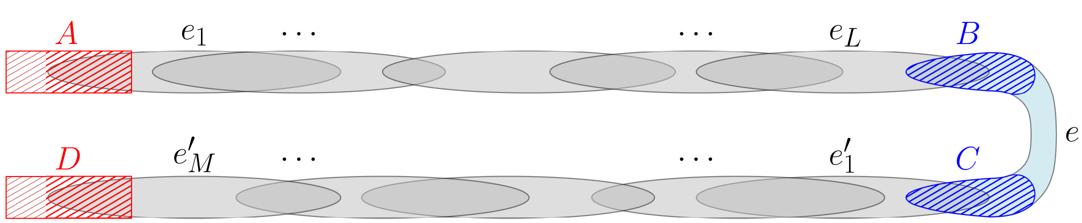

Fig. 9: Illustration of Lemma 3.4.

3.2 Archipelagos

In this subsection, we fix some $x^* \in V(\mathcal{H})$.

3.2.1 Definition

Definition 3.5. Let $\mathcal{I}$ and $\mathcal{I}'$ be disjoint islands in $\mathcal{H}$, where $\mathcal{I}'$ has an entry ε of size $k-1$. An edge $e \in E(\mathcal{H})$ of the form $e = \{x\} \cup \varepsilon$ for some $x \in V(\mathcal{I})$ is called a *crossing edge* from $\mathcal{I}$ to $\mathcal{I}'$. We denote by $C(\mathcal{I}, \mathcal{I}') \subseteq E(\mathcal{H})$ the set of all crossing edges from $\mathcal{I}$ to $\mathcal{I}'$ in $\mathcal{H}$. If $\mathcal{A}$ is a subhypergraph of $\mathcal{H}$ containing $\mathcal{I}$ and $\mathcal{I}'$, we use the notation $C_{\mathcal{A}}(\mathcal{I}, \mathcal{I}') := C(\mathcal{I}, \mathcal{I}') \cap E(\mathcal{A})$.

Remark. The above definition depends on the choice of ε (an island might have several possible entries suiting the definition). However, we will always specify the entries when defining islands and therefore consider crossing edges for those specific entries.

Definition 3.6. An x^*-*archipelago* is a subhypergraph $\mathcal{A}$ of $\mathcal{H}$ such that there exist subhypergraphs $\mathcal{I}_1, \dots, \mathcal{I}_N$ of $\mathcal{A}$ that are pairwise-disjoint islands with respective entries $\varepsilon_1, \dots, \varepsilon_N$ satisfying the following properties:

- $\varepsilon_1 = \{x^*\}$.
- $|\varepsilon_i| = k-1$ for all $2 \leq i \leq N$.
- $V(\mathcal{A}) = V(\mathcal{I}_1) \cup \ldots \cup V(\mathcal{I}_N)$.
- All edges in $E(\mathcal{A}) \setminus (E(\mathcal{I}_1) \cup \ldots \cup E(\mathcal{I}_N))$ are crossing edges between some of the $\mathcal{I}_i$, such that the digraph G defined by $V(G) = \{\mathcal{I}_1, \dots, \mathcal{I}_N\}$ and $E(G) = \{(\mathcal{I}_i, \mathcal{I}_j), C_{\mathcal{A}}(\mathcal{I}_i, \mathcal{I}_j) \neq \varnothing\}$ contains a spanning arborescence rooted at $\mathcal{I}_1$. If G is exactly a spanning arborescence rooted at $\mathcal{I}_1$, we say $\mathcal{A}$ is an *arborescent x^*-archipelago*.

Since x^* is fixed, we usually call $\mathcal{A}$ an *archipelago* for short.

Remark. By definition of a crossing edge, there cannot exist a crossing edge from some $\mathcal{I}_i$ to $\mathcal{I}_1$ in an archipelago since $|\varepsilon_1| = 1 \neq k-1$. In other words, $\mathcal{I}_1$ has in-degree zero in G.

Therefore, an archipelago is a union of pairwise-disjoint islands and crossing edges between some of them, satisfying specific properties. See Figure 10 for an example (for clarity, we will use $k = 3$ for all figures from now on). We will later see that an archipelago has a unique decomposition in islands, but for now we have to give ourselves islands and entries suiting the definition whenever we consider an archipelago.

3.2.2 Properties

The next two results show how $(k-2)$-linear paths in $\mathcal{A}$ are related to paths in the digraph G. Obviously, by definition of an archipelago, a $(k-2)$-linear path in $\mathcal{A}$ starting from x^* necessarily visits successive islands, using crossing edges to jump from one island to another. The following proposition states that, additionally, a crossing edge can only be used in one direction which is given by the digraph G, therefore each island is entered through its entry (hence the terminology) and it is impossible to reenter an island after leaving it.

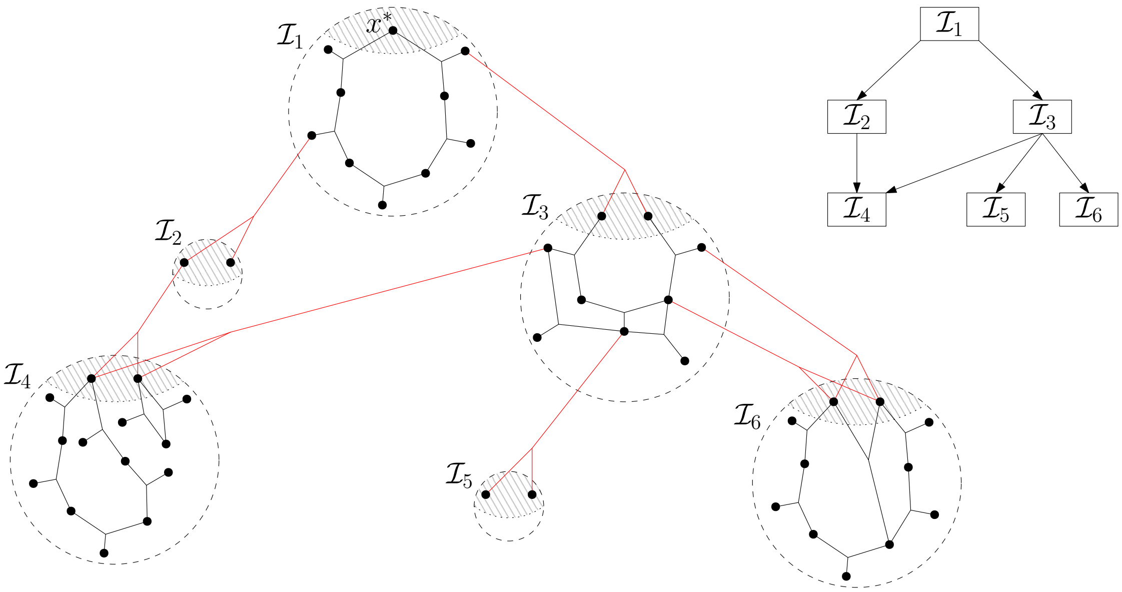

Fig. 10: An archipelago which is not arborescent (with the digraph G on the right). Crossing edges will always be represented in red.

Definition 3.7. Let G be a digraph and let $v, v' \in V(G)$. A *path* from v to v' in G is a sequence denoted by $v = v_1 \to v_2 \to \ldots \to v_l = v'$ ($l \geq 1$) where $v_1, \ldots, v_l \in V(G)$ are pairwise distinct and $(v_i, v_{i+1}) \in E(G)$ for all $1 \leq i \leq l-1$.

Proposition 3.8. *Let $\mathcal{A}$ be an archipelago, with $\mathcal{I}_1, \ldots, \mathcal{I}_N, \varepsilon_1, \ldots, \varepsilon_N, G$ suiting the definition. Let $\overrightarrow{P} = (e_1, \ldots, e_L)$ be a $(k-2)$-linear path from x^* to some $x \in V(\mathcal{I}_i)$ ($1 \leq i \leq N$) in $\mathcal{A}$. Then the islands visited by $\overrightarrow{P}$ form a path $\mathcal{I}_1 = \mathcal{I}_{i_1} \to \ldots \to \mathcal{I}_{i_M} = \mathcal{I}_i$ in G, and $\overrightarrow{P}$ is of the form $\overrightarrow{P} = \overrightarrow{P_1} \oplus (e_{1,2}) \oplus \overrightarrow{P_2} \oplus (e_{2,3}) \oplus \ldots \oplus \overrightarrow{P_{M-1}} \oplus (e_{M-1,M}) \oplus \overrightarrow{P_M}$ where:*

- *For all $1 \leq p \leq M$: $E(\overrightarrow{P_p}) \subseteq E(\mathcal{I}_{i_p})$.*
- *For all $2 \leq p \leq M$: $e_{p-1,p} \in C_{\mathcal{A}}(\mathcal{I}_{i_{p-1}}, \mathcal{I}_{i_p})$.*

In particular, if $L \geq 1$, then for all $1 \leq p \leq M$ there is an edge of $\overrightarrow{P}$ that contains ε_{i_p}.

Proof: That last assertion is clear: for $p = 1$ we have $\varepsilon_{i_p} = \varepsilon_1 = \{x^*\} \subset e_1$, and for $p \geq 2$ we have $\varepsilon_{i_p} \subset e_{p-1,p}$ by definition of $C_{\mathcal{A}}(\mathcal{I}_{i_{p-1}}, \mathcal{I}_{i_p})$. Let us now prove the main assertion.
We proceed by induction on L. The case $L = 0$ is trivial: we have $x = x^*$ so we can set $M = 1$ and $\overrightarrow{P_1} = \overrightarrow{P} = ()$. Let $L \geq 1$ and assume the result to be true for all $(k-2)$-linear paths that are shorter than $\overrightarrow{P}$. The idea is to separate two simple cases: either we are currently visiting the island $\mathcal{I}_i$ (case $e_L \in E(\mathcal{I}_i)$) or we have just jumped onto $\mathcal{I}_i$ from another island (case $e_L \notin E(\mathcal{I}_i)$). Let $y \in e_{L-1} \cap e_L$ if $L \geq 2$, or define $y = x^*$ if $L = 1$, so that in both cases $\overrightarrow{Q} := (e_1, \ldots, e_{L-1})$

is a $(k-2)$-linear path from x^* to y in $\mathcal{A}$. We have $y \in V(\mathcal{I}_j)$ for some $1 \leq j \leq N$. By the induction hypothesis, there exists a path $\mathcal{I}_1 = \mathcal{I}_{i_1} \to \ldots \to \mathcal{I}_{i_M} = \mathcal{I}_j$ in G such that we can write $\overrightarrow{Q} = \overrightarrow{Q_1} \oplus (e_{1,2}) \oplus \overrightarrow{Q_2} \oplus (e_{2,3}) \oplus \ldots \oplus \overrightarrow{Q_{M-1}} \oplus (e_{M-1,M}) \oplus \overrightarrow{Q_M}$ where $E(\overrightarrow{Q_p}) \subseteq E(\mathcal{I}_{i_p})$ for all $1 \leq p \leq M$ and $e_{p-1,p} \in C_{\mathcal{A}}(\mathcal{I}_{i_{p-1}}, \mathcal{I}_{i_p})$ for all $2 \leq p \leq M$.

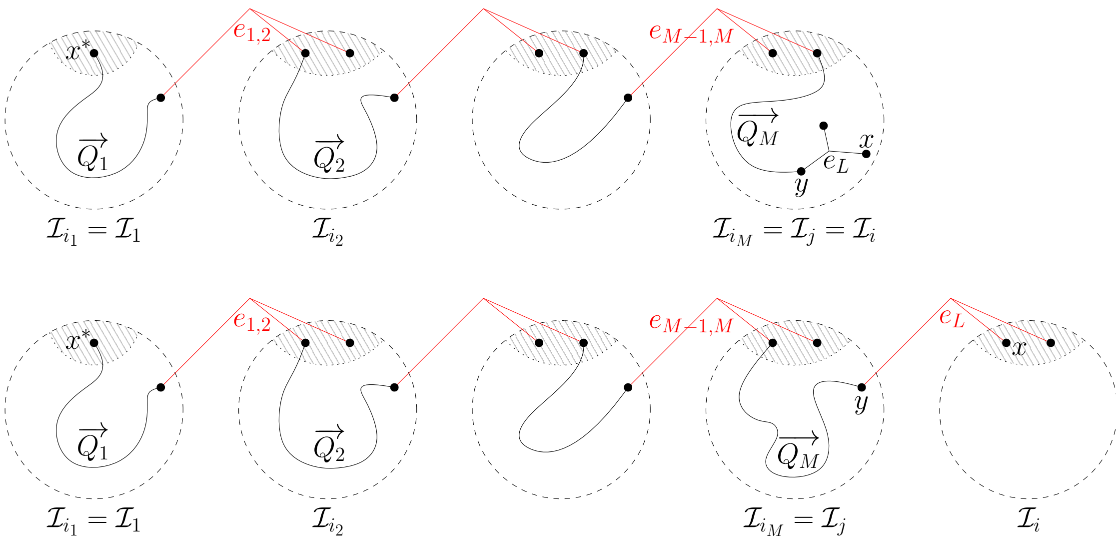

Fig. 11: Top: $e_L \in E(\mathcal{I}_i)$. Bottom: $e_L \notin E(\mathcal{I}_i)$.

- First suppose that $e_L \in E(\mathcal{I}_i)$ (see Figure 11, top). Since $y \in e_L$, this implies $i = j$, so $\overrightarrow{P_M} := \overrightarrow{Q_M} \oplus (e_L)$ satisfies $E(\overrightarrow{P_M}) \subseteq E(\mathcal{I}_i)$. Therefore, the following writing of $\overrightarrow{P}$ completes the proof: $\overrightarrow{P} = \overrightarrow{Q} \oplus (e_L) = \overrightarrow{Q_1} \oplus (e_{1,2}) \oplus \overrightarrow{Q_2} \oplus (e_{2,3}) \oplus \ldots \oplus \overrightarrow{Q_{M-1}} \oplus (e_{M-1,M}) \oplus \overrightarrow{P_M}$.

- Now suppose $e_L \notin E(\mathcal{I}_i)$ (see Figure 11, bottom), then by definition of an archipelago we have either $e_L \in C_{\mathcal{A}}(\mathcal{I}_i, \mathcal{I}_j)$ or $e_L \in C_{\mathcal{A}}(\mathcal{I}_j, \mathcal{I}_i)$.
Suppose for a contradiction that $e_L \in C_{\mathcal{A}}(\mathcal{I}_i, \mathcal{I}_j)$ *i.e.* $e_L = \{x\} \cup \varepsilon_j$: in particular $j \neq 1$ (and $|\varepsilon_j| = k-1$), so the fact that $\varepsilon_j \subset e_{M-1,M}$ contradicts the $(k-2)$-linearity of $\overrightarrow{P}$ since $\varepsilon_j \subset e_L$.
Therefore $e_L \in C_{\mathcal{A}}(\mathcal{I}_j, \mathcal{I}_i)$. In particular $i \neq 1$ (and $|\varepsilon_i| = k-1$), so it is impossible that $\mathcal{I}_i$ has been visited before: if we had $i \in \{i_1, \ldots, i_M\}$ then some edge of $\overrightarrow{Q}$ would contain ε_i which would contradict the $(k-2)$-linearity of $\overrightarrow{P}$ once again. Setting $i_{M+1} := i$, this ensures that the islands visited by $\overrightarrow{P}$ form a path $\mathcal{I}_1 = \mathcal{I}_{i_1} \to \ldots \to \mathcal{I}_{i_M} = \mathcal{I}_j \to \mathcal{I}_{i_{M+1}} = \mathcal{I}_i$ in G, and we can write $\overrightarrow{P} = \overrightarrow{Q} \oplus (e_{M,M+1}) \oplus \overrightarrow{P_{M+1}}$ where $e_{M,M+1} := e_L \in C_{\mathcal{A}}(\mathcal{I}_{i_M}, \mathcal{I}_{i_{M+1}})$ and $\overrightarrow{P_{M+1}} := ()$, which concludes. □

Conversely, paths in G yield $(k-2)$-linear paths in $\mathcal{A}$. The following proposition is a generalization to archipelagos of the property that defines an island.

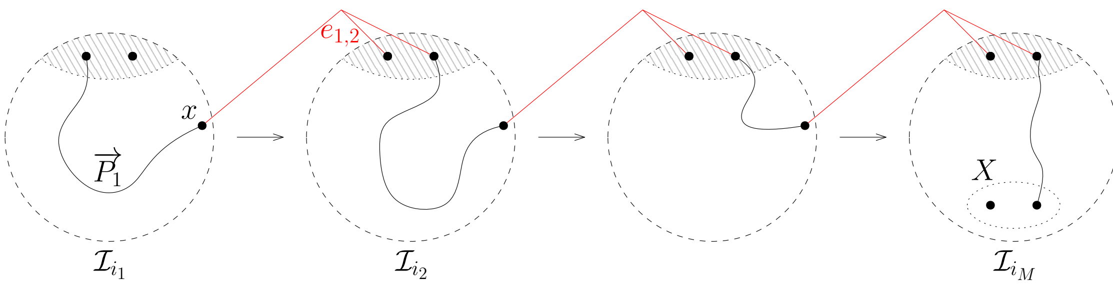

Fig. 12: An (ε_{i_1}, X)-extendable path in an archipelago.

Proposition 3.9. *Let $\mathcal{A}$ be an archipelago, with $\mathcal{I}_1, \dots, \mathcal{I}_N, \varepsilon_1, \dots, \varepsilon_N, G$ suiting the definition. Let $X \subset V(\mathcal{A})$ such that $1 \le |X| \le k-1$ and $X \notin \{\varepsilon_2, \dots, \varepsilon_N\}$. For all $1 \le j \le N$ and for every path $\mathcal{I}_j = \mathcal{I}_{i_1} \to \dots \to \mathcal{I}_{i_M}$ in G satisfying $X \cap V(\mathcal{I}_{i_M}) \neq \varnothing$ and $X \cap V(\mathcal{I}_{i_p}) = \varnothing$ for all $1 \le p \le M-1$, there exists an (ε_j, X)-extendable path $\overrightarrow{P}$ in $\mathcal{A}$ of the form $\overrightarrow{P} = \overrightarrow{P_1} \oplus (e_{1,2}) \oplus \overrightarrow{P_2} \oplus (e_{2,3}) \oplus \dots \oplus \overrightarrow{P_{M-1}} \oplus (e_{M-1,M}) \oplus \overrightarrow{P_M}$ where:*

- *For all $1 \le p \le M$: $E(\overrightarrow{P_p}) \subseteq E(\mathcal{I}_{i_p})$.*
- *For all $2 \le p \le M$: $e_{p-1,p} \in C_{\mathcal{A}}(\mathcal{I}_{i_{p-1}}, \mathcal{I}_{i_p})$.*

Proof: We proceed by induction on M.

- First suppose $M = 1$: we need to show that if $X \cap V(\mathcal{I}_j) \neq \varnothing$ then there exists an (ε_j, X)-extendable path in $\mathcal{I}_j$. This is basically the definition of an island, except that X is not necessarily entirely included in $V(\mathcal{I}_j)$. This is not a problem: since $X \notin \{\varepsilon_2, \dots, \varepsilon_N\}$ by assumption, there exists an $(\varepsilon_j, X \cap V(\mathcal{I}_j))$-extendable path $\overrightarrow{P}$ in $\mathcal{I}_j$ by definition of an island, and $\overrightarrow{P}$ is also (ε_j, X)-extendable by Lemma 3.3.

- Now suppose $M \ge 2$ and assume the result to be true for all shorter paths in G. We build the desired (ε_{i_1}, X)-extendable path by assembling three parts:

  (1) By the induction hypothesis, there exists an (ε_{i_2}, X)-extendable path $\overrightarrow{P'}$ in $\mathcal{A}$ of the form $\overrightarrow{P'} = \overrightarrow{P_2} \oplus (e_{2,3}) \oplus \overrightarrow{P_3} \oplus (e_{3,4}) \oplus \dots \oplus \overrightarrow{P_{M-1}} \oplus (e_{M-1,M}) \oplus \overrightarrow{P_M}$ where $E(\overrightarrow{P_p}) \subseteq E(\mathcal{I}_{i_p})$ for all $2 \le p \le M$ and $e_{p-1,p} \in C_{\mathcal{A}}(\mathcal{I}_{i_{p-1}}, \mathcal{I}_{i_p})$ for all $3 \le p \le M$.

  (2) Let $e_{1,2} \in C_{\mathcal{A}}(\mathcal{I}_{i_1}, \mathcal{I}_{i_2})$, which exists since $(\mathcal{I}_{i_1}, \mathcal{I}_{i_2}) \in E(G)$: we have $e_{1,2} = \{x\} \cup \varepsilon_{i_2}$ for some $x \in V(\mathcal{I}_{i_1})$.

  (3) Finally, by definition of an island, there exists an (ε_{i_1}, x)-extendable path $\overrightarrow{P_1}$ in $\mathcal{I}_{i_1}$.

The path $\overrightarrow{P} := \overrightarrow{P_1} \oplus (e_{1,2}) \oplus \overrightarrow{P'}$ is represented in Figure 12. Lemma 3.4 applied to $A = \varepsilon_{i_1}$, $B = \{x\}$, $C = \varepsilon_{i_2}$ and $D = X$ ensures that $\overrightarrow{P}$ is an (ε_{i_1}, X)-extendable path. □

We get the following characterization for the entries of an archipelago:

Proposition 3.10. *Let $\mathcal{A}$ be an archipelago, with $\mathcal{I}_1, \ldots, \mathcal{I}_N, \varepsilon_1, \ldots, \varepsilon_N$ suiting the definition. Let $X \subseteq V(\mathcal{A})$ such that $1 \leq |X| \leq k-1$. There exists an (x^*, X)-extendable path in $\mathcal{A}$ if and only if $X \notin \{\varepsilon_2, \ldots, \varepsilon_N\}$.*

Proof: We distinguish both cases:

- Suppose $X = \varepsilon_i$ for some $2 \leq i \leq N$. Let $\overrightarrow{P}$ be a $(k-2)$-linear path from x^* to ε_i in $\mathcal{A}$, then $\overrightarrow{P}$ is a $(k-2)$-linear path from x^* to x in $\mathcal{A}$ for some $x \in \varepsilon_i$. By Proposition 3.8, some edge of $\overrightarrow{P}$ (necessarily the last one, since $\overrightarrow{P}$ is from x^* to ε_i) contains ε_i, which proves that $\overrightarrow{P}$ is not (x^*, ε_i)-extendable.
- Suppose $X \notin \{\varepsilon_2, \ldots, \varepsilon_N\}$. Out of all the paths in G from $\mathcal{I}_1$ to one of the islands intersecting X (recall that G contains a spanning arborescence rooted at $\mathcal{I}_1$, so there exists at least one), consider a shortest one, so that X only intersects the last island of that path. We can now apply Proposition 3.9: there exists an (ε_1, X)-extendable path in $\mathcal{A}$, which concludes since $\varepsilon_1 = \{x^*\}$. □

Corollary 3.11. *Let $\mathcal{A}$ be an archipelago in $\mathcal{H}$. For all $x \in V(\mathcal{A})$, there exists a $(k-2)$-linear path from x^* to x in $\mathcal{A}$. In particular, $V(\mathcal{A}) \subseteq LCC_{\mathcal{H}}^{k-2}(x^*)$.*

Proof: Let $x \in V(\mathcal{A})$: applying Proposition 3.10 to $X = \{x\}$ shows that there exists a $(k-2)$-linear path from x^* to x in $\mathcal{A}$. □

Finally, we show that an archipelago has a unique decomposition.

Proposition 3.12. *Any archipelago $\mathcal{A}$ has unique islands and entries suiting the definition.*

Proof: Let $\varepsilon_1, \ldots, \varepsilon_N$ be entries suiting the definition: we have $\varepsilon_1 = \{x^*\}$, moreover $\{\varepsilon_2, \ldots, \varepsilon_N\}$ is exactly the set of all subsets $X \subset V(\mathcal{A})$ such that $1 \leq |X| \leq k-1$ and there exists no (x^*, X)-extendable path in $\mathcal{A}$ by Proposition 3.10, so these entries are unique. Suppose for a contradiction that $\{\mathcal{I}_1, \ldots, \mathcal{I}_N\}$ and $\{\mathcal{I}'_1, \ldots, \mathcal{I}'_N\}$ are two distinct sets of islands suiting the definition, where $\mathcal{I}_i$ and $\mathcal{I}'_i$ have the same entry ε_i for all $1 \leq i \leq N$. Since islands are induced subhypergraphs of $\mathcal{A}$, $\{V(\mathcal{I}_1), \ldots, V(\mathcal{I}_N)\}$ and $\{V(\mathcal{I}'_1), \ldots, V(\mathcal{I}'_N)\}$ are two distinct partitions of $V(\mathcal{A})$, so there exists $1 \leq i \neq j \leq N$ such that $V(\mathcal{I}_i) \cap V(\mathcal{I}'_j) \neq \varnothing$. Let $x \in V(\mathcal{I}_i) \cap V(\mathcal{I}'_j)$.

- Using the first decomposition, there exists an (ε_i, x)-extendable path $\overrightarrow{P} = (e_1, \ldots, e_L)$ in $\mathcal{I}_i$ by definition of an island. For all $2 \leq l \leq N$, no edge of $\overrightarrow{P}$ contains ε_l: if $l = i$ then this is the definition of an (ε_i, x)-extendable path, and if $l \neq i$ then this is obvious since $V(\mathcal{I}_i)$ is disjoint from ε_l.
- Using the second decomposition, since $x \in V(\mathcal{I}'_j)$ and ε_i is disjoint from $V(\mathcal{I}'_j)$, we can define $r := \inf\{1 \leq p \leq L \text{ such that } e_p \not\subset V(\mathcal{I}'_j)\}$. We have $e_r \not\subset V(\mathcal{I}'_j)$, however e_r intersects $V(\mathcal{I}'_j)$ by minimality of r, therefore e_r is necessarily a crossing edge for the second decomposition. This means that $\varepsilon_l \subset e_r$ for some $2 \leq l \leq N$, which contradicts what we have just established. $\square$

Notation 3.13. Let $\mathcal{A}$ be an archipelago. Proposition 3.12 allows us to define without ambiguity:
- $\mathcal{I}(\mathcal{A})$: the set of islands of $\mathcal{A}$.
- $\varepsilon(\mathcal{A})$: the set of entries of the islands of $\mathcal{A}$.
- $G(\mathcal{A})$: the digraph from the definition of an archipelago.

4 $(k-2)$-linear connected components: structure and computation

In this section, we suppose again that $\mathcal{H}$ is k-uniform and we fix some $x^* \in V(\mathcal{H})$.

4.1 Main results

Our two main results about $(k-2)$-linear connected components, one structural and the other algorithmic, can be assembled into the following main theorem which will be proven in this section.

Definition 4.1. An x^*-archipelago $\mathcal{A}$ in $\mathcal{H}$ is said to be maximal if there is no x^*-archipelago in $\mathcal{H}$ that has $\mathcal{A}$ as a strict subhypergraph.

Theorem 4.2. $\mathcal{H}[LCC^{k-2}_{\mathcal{H}}(x^*)]$ *is the unique maximal* x^**-archipelago in* $\mathcal{H}$*, and it can be computed in* $O(m^2k)$ *time where* $m = |E(\mathcal{H})|$.

Corollary 4.3. *For all* $k \geq 3$*,* $\textsc{HypConnectivity}_{k,k-2}$ *is solvable in polynomial time.*

4.2 The key intermediate result

Theorem 4.2 will come as a straightforward consequence of the following theorem, which is illustrated in Figure 13:

Theorem 4.4. *There exists an* x^**-archipelago* $\mathcal{A}$ *in* $\mathcal{H}$ *and a partition* $E(\mathcal{H}) = E(\mathcal{A}) \cup E_{cut} \cup E_{ext}$ *(where* E_{cut} *and/or* E_{ext} *may be empty) such that:*

(1) Every $e \in E_{cut}$ *is of the form* $e = \varepsilon \cup \{x\}$ *for some entry* ε *of* $\mathcal{A}$ *of size* $k-1$ *and some* $x \notin V(\mathcal{A})$*;*

(2) Every $e \in E_{ext}$ *is disjoint from* $V(\mathcal{A})$*.*

Moreover, this partition can be computed in $O(m^2 k)$ time where $m = |E(\mathcal{H})|$.

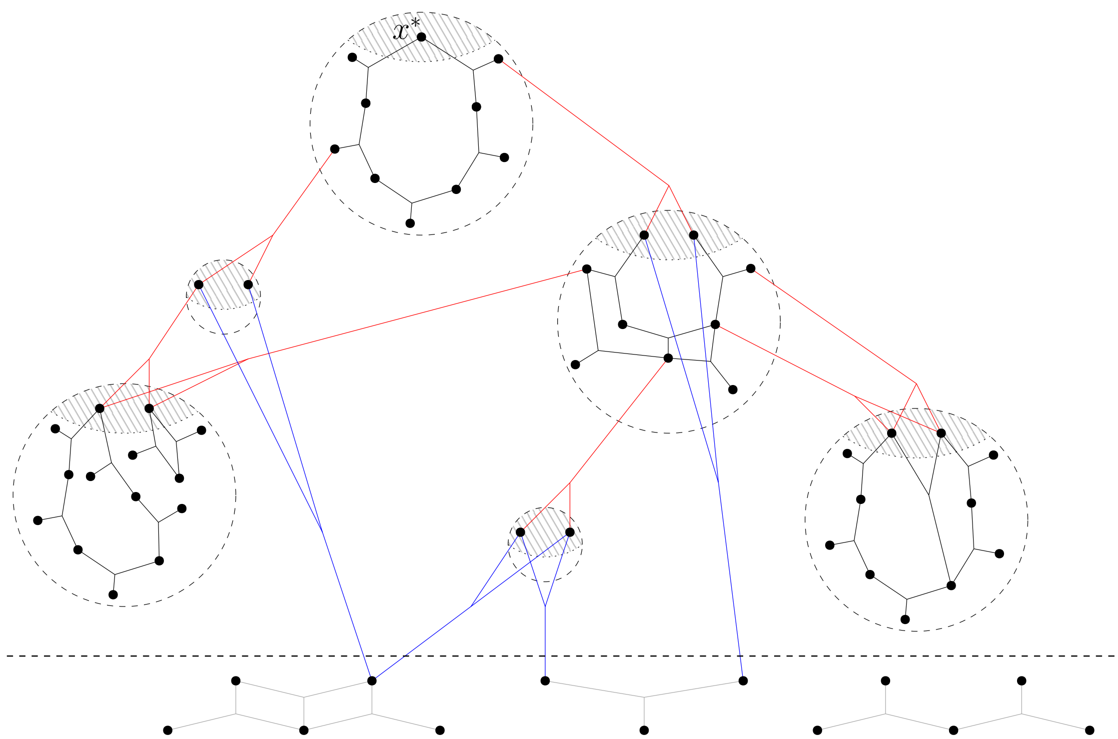

Fig. 13: The hypergraph $\mathcal{H}$ is represented in full. In black and red: $E(\mathcal{A})$ (archipelago). In blue: E_{cut}. In grey, below the dashed line: E_{ext}.

4.2.1 Augmenting archipelagos

Our algorithm proving Theorem 4.4 will build the archipelago $\mathcal{A} = \mathcal{H}[LCC_{\mathcal{H}}^{k-2}(x^*)]$ edge by edge until reaching maximality, and then throw the remaining edges into E_{cut} and E_{ext}. Therefore, we need to address the following question: given an archipelago $\mathcal{A}$ and an edge $e \in E(\mathcal{H}) \setminus E(\mathcal{A})$, is $\mathcal{A} \cup e$ an archipelago (and if so, for what decomposition)? Here $\mathcal{A} \cup e$ denotes the subhypergraph of $\mathcal{H}$ defined by $V(\mathcal{A} \cup e) = V(A) \cup e$ and $E(\mathcal{A} \cup e) = E(A) \cup \{e\}$. The answer will depend on the way e intersects $\mathcal{A}$:

Definition 4.5. Let $\mathcal{A}$ be an archipelago. An edge $e \in E(\mathcal{H}) \setminus E(\mathcal{A})$ is of one of five $\mathcal{A}$-*types*:

1. "exterior": $|e \cap V(\mathcal{A})| = 0$.
2. "new crossing": $|e \cap V(\mathcal{A})| = 1$.
3. "crossing": e is a crossing edge between two islands of $\mathcal{A}$.
4. "cut": e is of the form $e = \varepsilon \cup \{x\}$, where ε is an entry of $\mathcal{A}$ of size $k-1$ and $x \in V(\mathcal{H}) \setminus V(\mathcal{A})$.

5. "other": e is none of the above.

Those are well defined because the islands and entries of an archipelago are unique by Proposition 3.12. The five $\mathcal{A}$-types are illustrated in Figure 14.

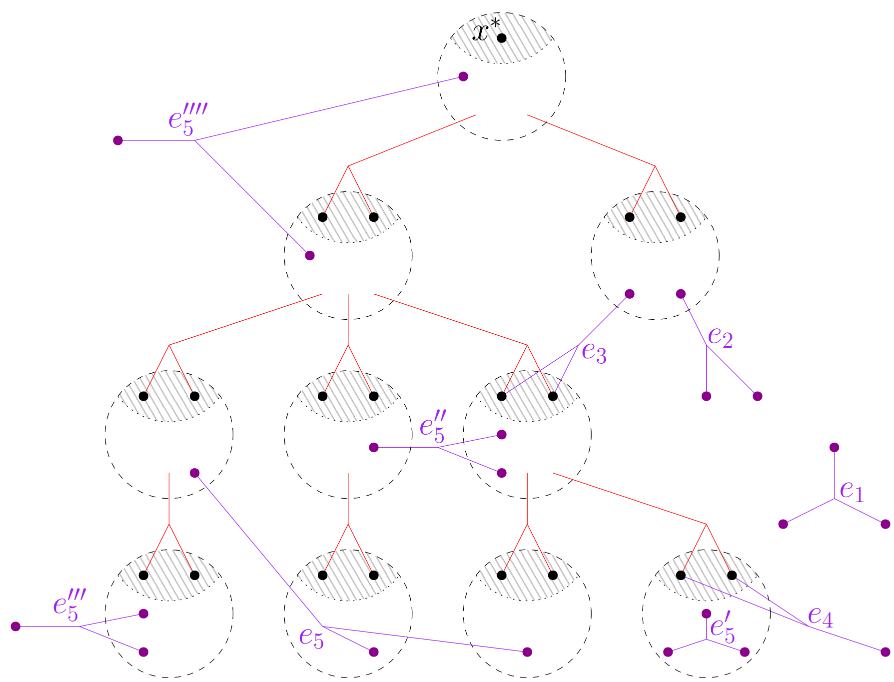

Fig. 14: An arborescent archipelago $\mathcal{A}$ (the inside of the islands is not detailed), and some edges in $E(\mathcal{H}) \setminus E(\mathcal{A})$ (in purple). The names of the edges follow the numbering from Definition 4.5: e_1 is of $\mathcal{A}$-type "exterior", e_2 is of $\mathcal{A}$-type "new crossing", etc.

Fundamentally:

- The $\mathcal{A}$-types "crossing", "new crossing" and "other" correspond to edges that get added to the archipelago.
- The $\mathcal{A}$-type "cut" corresponds to E_{cut}.
- The $\mathcal{A}$-type "exterior" corresponds to E_{ext}.

Let $\mathcal{A}$ be an archipelago, with islands $\mathcal{I}_1, \ldots, \mathcal{I}_N$ and entries $\varepsilon_1, \ldots, \varepsilon_N$, and let $e \in E(\mathcal{H}) \setminus E(\mathcal{A})$. We now explain why $\mathcal{A} \cup e$ is an archipelago if e is of $\mathcal{A}$-type "crossing", "new crossing" or "other". In the case of the $\mathcal{A}$-types "new crossing" and "other", the arborescent nature of the archipelago will be preserved, so those edges will be added first in our algorithm so that the archipelago remains arborescent for as long as possible. Even though the decomposition of $\mathcal{A} \cup e$ is given by $\mathcal{I}(A \cup e)$ and $\varepsilon(A \cup e)$ alone, we also describe $G(\mathcal{A} \cup e)$ in the arborescent case.

I) e is of $\mathcal{A}$-type "new crossing"

This case is easy: a new island is created, with e being the crossing edge that connects it to the rest (see Figure 15).

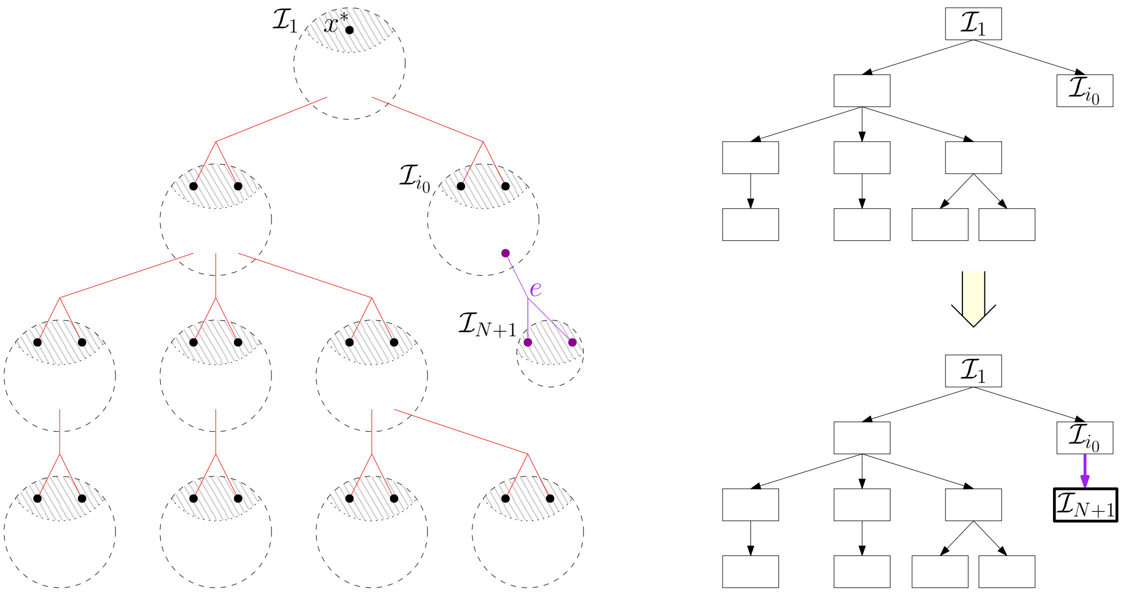

Fig. 15: The archipelago $\mathcal{A} \cup e$ where $\mathcal{A}$ is as in Figure 14 and $e = e_2$. On the right: the arborescences $G(\mathcal{A})$ (top) and $G(\mathcal{A} \cup e)$ (bottom).

Proposition 4.6. *Suppose $\mathcal{A}$ is arborescent and e is of $\mathcal{A}$-type "new crossing". Let $1 \leq i_0 \leq N$ be the index of the only island that intersects e, and let $\mathcal{I}_{N+1}$ be the empty island with entry $\varepsilon_{N+1} := e \setminus V(\mathcal{I}_{i_0})$. Then $\mathcal{A} \cup e$ is an arborescent archipelago with:*
- $\mathcal{I}(A \cup e) = \mathcal{I}(A) \cup \{\mathcal{I}_{N+1}\}$.
- $\varepsilon(\mathcal{A} \cup e) = \varepsilon(\mathcal{A}) \cup \{\varepsilon_{N+1}\}$.
- *$G(\mathcal{A} \cup e)$ defined as the digraph obtained from $G(\mathcal{A})$ by adding a new vertex $\mathcal{I}_{N+1}$ and an arc $(\mathcal{I}_{i_0}, \mathcal{I}_{N+1})$.*

Proof: This is clear: e is a crossing edge from $\mathcal{I}_{i_0}$ to $\mathcal{I}_{N+1}$, hence the new arc in $G(\mathcal{A} \cup e)$ which is obviously an arborescence since $G(\mathcal{A})$ is. $\square$

II) e is of $\mathcal{A}$-type "other"

By definition, this means that: $|e \cap V(\mathcal{A})| \geq 2$, e is not a crossing edge, and e is not of the form $\varepsilon \cup \{x\}$ where ε is an entry of $\mathcal{A}$ of size $k-1$ and $x \in V(\mathcal{H}) \setminus V(\mathcal{A})$.

This case is more complicated. Consider Figure 14. If e only intersects one island ($e = e_5'$ or $e = e_5'''$ for instance), then it should be easy to show that this island plus e is still an island. If e links several islands however, then the way to redefine islands is not as straightforward, since e is not a crossing edge. Suppose $e = e_5$ for instance, as in Figure 16. The fact that e acts as a bridge between several islands creates new paths: for example, we have an (x^*, ε_6)-extendable path in $\mathcal{A} \cup e$ (represented schematically in Figure 16), therefore ε_6 would not be an entry of $\mathcal{A} \cup e$ (recall Proposition 3.10). Actually, it can be shown that the subhypergraph $\mathcal{I}$, formed by the union of $\mathcal{I}_2, \mathcal{I}_4, \mathcal{I}_5, \mathcal{I}_6, \mathcal{I}_8, \mathcal{I}_9$ and the crossing edges between them as well as e, is an island with entry ε_2. Therefore, $\mathcal{A} \cup e$ is an archipelago with five islands: $\mathcal{I}_1, \mathcal{I}_3, \mathcal{I}_7, \mathcal{I}_{10}, \mathcal{I}$. On this example, we see how adding en edge can merge islands together. We are now going to generalize this argument.

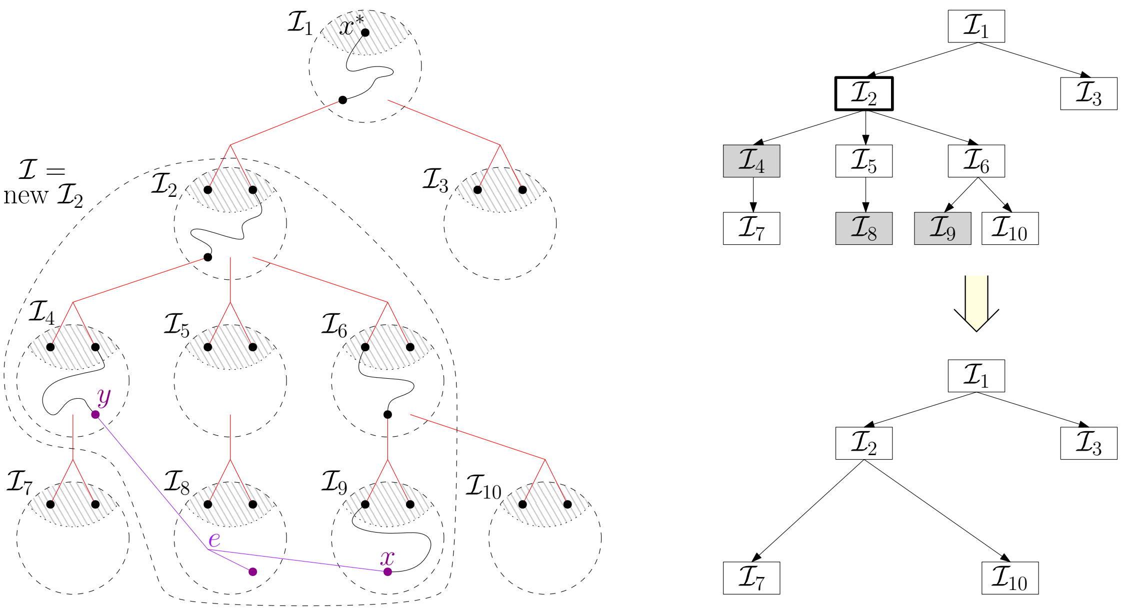

Fig. 16: The archipelago $\mathcal{A} \cup e$ where $\mathcal{A}$ is as in Figure 14 and $e = e_5$. On the right: the arborescences $G(\mathcal{A})$ (top) and $G(\mathcal{A} \cup e)$ (bottom).

Definition 4.7. Let G be an arborescence rooted at some $v^* \in V(G)$, and let $U = \{v_1, \ldots, v_r\} \subseteq V(G)$. For all $1 \leq i \leq r$, let $v^* = v_{i,1} \to \ldots \to v_{i,l_i} = v_i$ be the unique path from v^* to v_i in G. Define $i_0 := \sup\{1 \leq p \leq \min_{1 \leq i \leq r} l_i \mid v_{1,p} = \ldots = v_{r,p}\}$. The *lowest common ancestor of U in G* is defined as $\mathrm{LCA}_G(U) := v_{i_0}$.

Definition 4.8. Let G be an arborescence and let $v \in V(G)$. For all $1 \leq i \leq r$, let $v = v_{i,1} \to \ldots \to v_{i,l_i} = v_i$ be a path from v to some $v_i \in V(G)$ in G. Let $U := \bigcup_{1 \leq i \leq r} \{v_{i,1}, \ldots, v_{i,l_i}\}$ be the set of all vertices on these paths. *Merging U into v* means:

- deleting all vertices in $U \setminus \{v\}$;

- deleting all arcs between vertices in U;
- replacing every arc $(u, w) \in (U \setminus \{v\}) \times (V(G) \setminus U)$ by an arc (v, w).

Example. Figure 16 features a merging process on the right. The three considered paths are: $\mathcal{I}_2 \leftarrow \mathcal{I}_4$, $\mathcal{I}_2 \leftarrow \mathcal{I}_5 \leftarrow \mathcal{I}_8$, $\mathcal{I}_2 \leftarrow \mathcal{I}_6 \leftarrow \mathcal{I}_9$. The set $U = \{\mathcal{I}_2, \mathcal{I}_4, \mathcal{I}_5, \mathcal{I}_6, \mathcal{I}_8, \mathcal{I}_9\}$ has been merged into $v = \mathcal{I}_2$.

Proposition 4.9. *Suppose $\mathcal{A}$ is arborescent and e is of $\mathcal{A}$-type "other". Define:*

- $J_0 := \{1 \leq i \leq N$ *such that* $V(\mathcal{I}_i) \cap e \neq \varnothing\}$*, the set of indices of the islands that e intersects.*
- i_0 *the index such that* $\mathcal{I}_{i_0} := \mathrm{LCA}_{G(\mathcal{A})}(\{\mathcal{I}_i, i \in J_0\})$.
- $J := \bigcup_{i \in J_0} \{1 \leq j \leq N \mid \mathcal{I}_j$ *is on the path from* $\mathcal{I}_{i_0}$ *to* $\mathcal{I}_i$ *in* $G(\mathcal{A})\} \supseteq J_0$.
- $\mathcal{I} := \mathcal{A}[\bigcup_{j \in J} V(\mathcal{I}_j)]$*, the island that will replace $\mathcal{I}_{i_0}$ (with the same entry ε_{i_0}).*

Then $\mathcal{A} \cup e$ is an arborescent archipelago with:

- $\mathcal{I}(\mathcal{A} \cup e) = (\mathcal{I}(\mathcal{A}) \setminus \{\mathcal{I}_j, j \in J\}) \cup \{\mathcal{I}\}$.
- $\varepsilon(\mathcal{A} \cup e) = \varepsilon(\mathcal{A}) \setminus \{\varepsilon_j, j \in J \setminus \{i_0\}\}$.
- $G(\mathcal{A} \cup e)$ *defined as the digraph obtained from* $G(\mathcal{A})$ *by merging* $\{\mathcal{I}_j, j \in J\}$ *into* $\mathcal{I}_{i_0}$.

Proof: For visual help, refer to Figure 16: in this example we have $J_0 = \{4, 8, 9\}$, $i_0 = 2$, $J = \{2, 4, 5, 6, 8, 9\}$. The merging process that defines $G(\mathcal{A} \cup e)$ clearly preserves the fact that the digraph is an arborescence. To complete the proof, we only need to show that $\mathcal{I}$ is an island with entry ε_{i_0}: let $X \subset V(\mathcal{I})$ such that $1 \leq |X| \leq k-1$ (and $X \neq \varepsilon_{i_0}$ if $i_0 \neq 1$), we need to find an (ε_{i_0}, X)-extendable path in $\mathcal{I}$. As visible in Figure 14, e might or might not be included in $V(\mathcal{A})$, so in general we have $V(\mathcal{I}) = \bigcup_{j \in J} V(\mathcal{I}_j) \cup e$. We distinguish four possibilities:

1) <u>Case 1</u>: $X \subset \bigcup_{j \in J} V(\mathcal{I}_j)$ and $X \notin \{\varepsilon_j, j \in J \setminus \{i_0\}\}$.
Of all paths in $G(\mathcal{A})$ from $\mathcal{I}_{i_0}$ to an island intersecting X, let $\mathcal{I}_{i_0} = \mathcal{I}_{j_1} \to \ldots \to \mathcal{I}_{j_M}$ be a shortest one, so that $X \cap V(\mathcal{I}_{j_M}) \neq \varnothing$ and $X \cap V(\mathcal{I}_{j_p}) = \varnothing$ for all $1 \leq p \leq M-1$. Note that, by definition of J, we have $\{j_1, \ldots, j_M\} \subseteq J$, so the islands $\mathcal{I}_{j_1}, \ldots, \mathcal{I}_{j_M}$ are all subhypergraphs of $\mathcal{I}$ and all crossing edges between them in $\mathcal{A}$ are edges of $\mathcal{I}$. By Proposition 3.9, there exists an (ε_{i_0}, X)-extendable path $\overrightarrow{P}$ in $\mathcal{A}$ such that $E(\overrightarrow{P}) \subseteq \bigcup_{p=1}^{M} E(\mathcal{I}_{j_p}) \cup \bigcup_{p=2}^{M} C_{\mathcal{A}}(\mathcal{I}_{j_{p-1}}, \mathcal{I}_{j_p}) \subseteq E(\mathcal{I})$, which concludes.

2) <u>Case 2</u>: X intersects both $\bigcup_{j \in J} V(\mathcal{I}_j)$ and $e \setminus \bigcup_{j \in J} V(\mathcal{I}_j)$.
Define $X' := X \cap \bigcup_{j \in J} V(\mathcal{I}_j)$, we have $1 \leq |X'| \leq k-1$. Case 1 applied to X' gives us an (ε_{i_0}, X')-extendable path $\overrightarrow{P}$ in $\mathcal{I}$, which is also (ε_{i_0}, X)-extendable by Lemma 3.3 applied to $A = \varepsilon_{i_0}$, $B = X'$ and $B' = X$.

3) <u>Case 3</u>: $X \subset e \setminus \bigcup_{j \in J} V(\mathcal{I}_j)$.
Define $X' := e \cap \bigcup_{j \in J} V(\mathcal{I}_j)$, we have $2 \leq |X'| \leq k-1$ hence $1 \leq |X| \leq k-2$: indeed $|X'| \geq 2$ by definition of the $\mathcal{A}$-type "other", and $|X'| \leq k-1$ because $e \setminus \bigcup_{j \in J} V(\mathcal{I}_j) \supseteq X \neq \varnothing$. Moreover $X' \notin \{\varepsilon_j, j \in J \setminus \{i_0\}\}$, otherwise e would be of $\mathcal{A}$-type "cut". We can thus apply Case 1 to X', which gives us an (ε_{i_0}, X')-extendable path $\overrightarrow{P}$ in $\mathcal{I}$. Lemma 3.4 applied to

$A = \varepsilon_{i_0}$, $B = X'$, $C = D = X$ and $\overrightarrow{Q} = ()$ ensures that $\overrightarrow{P} \oplus (e)$ is an (ε_{i_0}, X)-extendable path in $\mathcal{I}$.

4) Case 4: $X = \varepsilon_j$ for some $j \in J \setminus \{i_0\}$.
In particular $|J| \geq 2$, so e intersects several islands. Note that, since $\mathcal{I}_{i_0}$ is a strict ancestor of $\mathcal{I}_j$ in $G(\mathcal{A})$, we have $j \neq 1$. Remember our example from Figure 16: we considered $X = \varepsilon_6$, and the (ε_2, X)-extendable path was obtained by going from ε_2 to $e \cap V(\mathcal{I}_4) = \{y\}$, then using e to jump from $\mathcal{I}_4$ to $\mathcal{I}_9$, then going from $e \cap V(\mathcal{I}_9) = \{x\}$ to X. Let us now build this path in general.

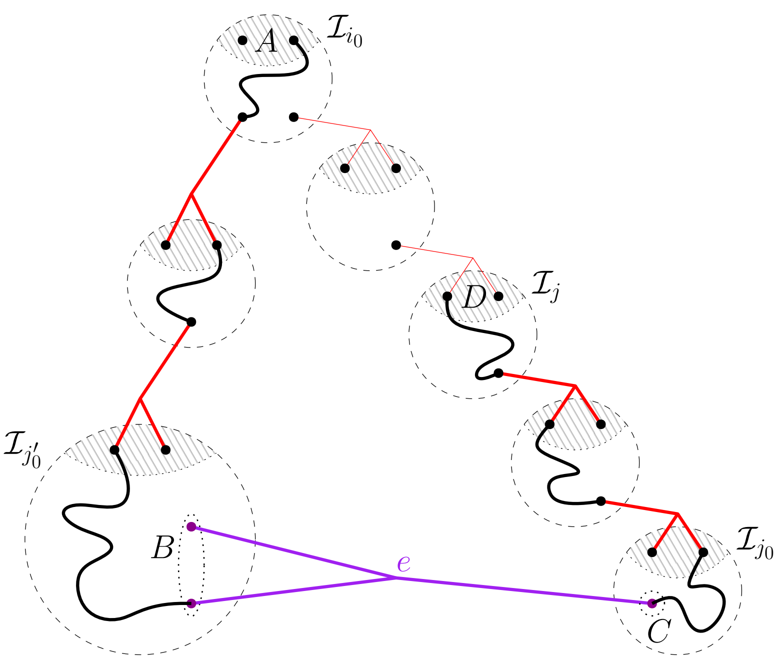

Fig. 17: Illustration of Case 4 from Proposition 4.9. The bold paths (in red and black) are $\overrightarrow{P}$ on the right and $\overrightarrow{Q}$ on the left.

- Let $j_0 \in J_0$ such that the path $\mathcal{I}_j = \mathcal{I}_{i_1} \to \ldots \to \mathcal{I}_{i_M} = \mathcal{I}_{j_0}$ in $G(\mathcal{A})$ is shortest, so that $i_p \notin J_0$ for all $1 \leq p \leq M-1$. This means $e \cap V(\mathcal{I}_{i_M}) \neq \varnothing$ and $e \cap V(\mathcal{I}_{i_p}) = \varnothing$ for all $1 \leq p \leq M-1$. Since e intersects several islands, we know $1 \leq |e \cap V(\mathcal{I}_{j_0})| \leq k-1$. Moreover the fact that $j \neq 1$ implies that $j_0 \neq 1$, so $e \cap V(\mathcal{I}_{j_0}) \neq \varepsilon_{j_0}$, otherwise e would be of $\mathcal{A}$-type "crossing". We can thus apply Proposition 3.9 and get an $(\varepsilon_j, e \cap V(\mathcal{I}_{j_0}))$-extendable path $\overrightarrow{P}$ in $\mathcal{A}$ such that $E(\overrightarrow{P}) \subseteq \bigcup_{p=1}^{M} E(\mathcal{I}_{i_p}) \cup \bigcup_{p=2}^{M} C_{\mathcal{A}}(\mathcal{I}_{i_{p-1}}, \mathcal{I}_{i_p})$, hence $E(\overrightarrow{P}) \subseteq E(\mathcal{I})$ since $\{i_1, \ldots, i_M\} \subseteq J$ by definition of J. See Figure 17 (path on the right).
- Since the lowest common ancestor of $\{\mathcal{I}_i, i \in J_0\}$ is $\mathcal{I}_{i_0}$ and not $\mathcal{I}_j$, there exists $j_0' \in$

$J_0 \backslash \{j_0\}$ such that $\mathcal{I}_j$ is not an ancestor of $\mathcal{I}_{j_0'}$, so the path $\mathcal{I}_{i_0} = \mathcal{I}_{i_1'} \to \ldots \to \mathcal{I}_{i_{M'}'} = \mathcal{I}_{j_0'}$ from $\mathcal{I}_{i_0}$ to $\mathcal{I}_{j_0'}$ in $G(\mathcal{A})$ satisfies $\{i_1, \ldots, i_M\} \cap \{i_1', \ldots, i_{M'}'\} = \varnothing$ (see Figure 17 for the relative positions of the four islands in play: $\mathcal{I}_{i_0}$, $\mathcal{I}_j$, $\mathcal{I}_{j_0}$, $\mathcal{I}_{j_0'}$). As usual, we choose j_0' so that this path is shortest, this way we have $e \cap V(\mathcal{I}_{i_{M'}'}) \neq \varnothing$ and $e \cap V(\mathcal{I}_{i_p'}) = \varnothing$ for all $1 \leq p \leq M' - 1$. Since e intersects several islands, we know $1 \leq |e \cap V(\mathcal{I}_{j_0'})| \leq k - 1$. Moreover, if $j_0' \neq 1$ then $e \cap V(\mathcal{I}_{j_0'}) \neq \varepsilon_{j_0'}$ otherwise e would be of $\mathcal{A}$-type "crossing". We can thus apply Proposition 3.9 and get an $(\varepsilon_{i_0}, e \cap V(\mathcal{I}_{j_0'}))$-extendable path $\overrightarrow{Q}$ in $\mathcal{A}$ such that $E(\overrightarrow{Q}) \subseteq \bigcup_{p=1}^{M'} E(\mathcal{I}_{i_p'}) \cup \bigcup_{p=2}^{M'} C_{\mathcal{A}}(\mathcal{I}_{i_{p-1}'}, \mathcal{I}_{i_p'})$, hence $E(\overrightarrow{Q}) \subseteq E(\mathcal{I})$ since $\{i_1', \ldots, i_{M'}'\} \subseteq J$ by definition of J. See Figure 17 (path on the left).

- Let $\overrightarrow{P'}$ be the sequence obtained by reversing $\overrightarrow{P}$. Since $\overrightarrow{P}$ is an $(\varepsilon_j, e \cap V(\mathcal{I}_{j_0}))$-extendable path, $\overrightarrow{P'}$ is an $(e \cap V(\mathcal{I}_{j_0}), \varepsilon_j)$-extendable path. Lemma 3.4 applied to $A = X = \varepsilon_{i_0}$, $B = e \cap V(\mathcal{I}_{j_0'})$, $C = e \cap V(\mathcal{I}_{j_0})$ and $D = \varepsilon_j$, whose conditions are fulfilled since $\{i_1, \ldots, i_M\} \cap \{i_1', \ldots, i_{M'}'\} = \varnothing$, ensures that $\overrightarrow{Q} \oplus (e) \oplus \overrightarrow{P'}$ is an $(\varepsilon_{i_0}, \varepsilon_j)$-extendable path in $\mathcal{I}$ which concludes. $\square$

III) e **is of** $\mathcal{A}$**-type "crossing"**

This is the easiest case: e is added as a crossing edge and the decomposition remains the same. Note that $\mathcal{A} \cup e$ might not be arborescent anymore (see $e = e_3$ from Figure 14 for example).

Proposition 4.10. *If e is of $\mathcal{A}$-type "crossing", then $\mathcal{A} \cup e$ is an archipelago with:*

- $\mathcal{I}(A \cup e) = \mathcal{I}(A)$.
- $\varepsilon(\mathcal{A} \cup e) = \varepsilon(\mathcal{A})$.

Proof: This is straightforward. $\square$

4.2.2 Formal algorithm

The algorithm Partition_Archipelago (Algorithm 1) returns a partition of the edges that satisfies Theorem 4.4. The procedures Add_NewCrossing, Add_Other and Add_Crossing (Algorithms 2 to 4) are nothing but algorithmic translations of Propositions 4.6, 4.9 and 4.10 respectively. Note that islands are simply implemented as vertex sets, because their edge sets are never used.

Algorithm 1 PARTITION_ARCHIPELAGO($\mathcal{H}, x^*$)

initialize $V(\mathcal{I}_1) \leftarrow \{x^*\}$
define $\varepsilon_1 \leftarrow \{x^*\}$
initialize the archipelago $\mathcal{A}$ with:
    $E(\mathcal{A}) \leftarrow \varnothing$
    $\mathcal{I}(\mathcal{A}) \leftarrow \{V(\mathcal{I}_1)\}$
    $\varepsilon(\mathcal{A}) \leftarrow \{\varepsilon_1\}$
    $G(\mathcal{A}) \leftarrow$ a digraph with only one vertex, labelled $\mathcal{I}_1$
initialize $N \leftarrow 1$ (index of the last created island)
while there exists $e \in E(\mathcal{H}) \setminus E(\mathcal{A})$ of $\mathcal{A}$-type "new crossing" or "other" **do**
    if e is of $\mathcal{A}$-type "new crossing" **then**
        update $\mathcal{A}$ as $\mathcal{A} \cup e$ by performing ADD_NEWCROSSING
    else
        update $\mathcal{A}$ as $\mathcal{A} \cup e$ by performing ADD_OTHER
    end if
end while
while there exists $e \in E(\mathcal{H}) \setminus E(\mathcal{A})$ of $\mathcal{A}$-type "crossing" **do**
    update $\mathcal{A}$ as $\mathcal{A} \cup e$ by performing ADD_CROSSING
end while
define $E_{cut} \leftarrow \{e \in E(\mathcal{H}) \setminus E(\mathcal{A}), e$ is of $\mathcal{A}$-type "cut"$\}$
define $E_{ext} \leftarrow \{e \in E(\mathcal{H}) \setminus E(\mathcal{A}), e$ is of $\mathcal{A}$-type "exterior"$\}$
return $E(\mathcal{A})$, E_{cut}, E_{ext}

Algorithm 2 ADD_NEWCROSSING

define $1 \leq i_0 \leq N$ as the only index such that $e \cap V(\mathcal{I}_{i_0}) \neq \varnothing$
initialize $V(\mathcal{I}_{N+1}) \leftarrow e \setminus V(\mathcal{I}_{i_0})$
define $\varepsilon_{N+1} \leftarrow e \setminus V(\mathcal{I}_{i_0})$
update the archipelago $\mathcal{A}$ as follows:
    $E(\mathcal{A}) \leftarrow E(\mathcal{A}) \cup \{e\}$
    $\mathcal{I}(\mathcal{A}) \leftarrow \mathcal{I}(\mathcal{A}) \cup \{V(\mathcal{I}_{N+1})\}$
    $\varepsilon(\mathcal{A} \cup e) \leftarrow \varepsilon(\mathcal{A}) \cup \{\varepsilon_{N+1}\}$
    $G(\mathcal{A} \cup e) \leftarrow$ the digraph obtained from $G(\mathcal{A})$ by adding a new vertex labelled $\mathcal{I}_{N+1}$ and an arc $(\mathcal{I}_{i_0}, \mathcal{I}_{N+1})$
$N \leftarrow N + 1$

Algorithm 3 ADD_OTHER

define $J_0 := \{1 \leq i \leq N$ such that $V(\mathcal{I}_i) \cap e \neq \varnothing\}$
define $1 \leq i_0 \leq N$ such that $\mathcal{I}_{i_0} = \mathrm{LCA}_{G(\mathcal{A})}(\{\mathcal{I}_i, i \in J_0\})$
define $J := \bigcup_{i \in J_0} \{1 \leq j \leq N$ such that $\mathcal{I}_j$ is on the path from $\mathcal{I}_{i_0}$ to $\mathcal{I}_i$ in $G(\mathcal{A})\}$
$V(\mathcal{I}_{i_0}) \leftarrow \bigcup_{i \in J} V(\mathcal{I}_j)$
update the archipelago $\mathcal{A}$ as follows:
    $E(\mathcal{A}) \leftarrow E(\mathcal{A}) \cup \{e\}$
    $\mathcal{I}(\mathcal{A}) \leftarrow \mathcal{I}(\mathcal{A}) \setminus \{V(\mathcal{I}_j), j \in J \setminus \{i_0\}\}$.
    $\varepsilon(\mathcal{A}) \leftarrow \varepsilon(\mathcal{A}) \setminus \{\varepsilon_j, j \in J \setminus \{i_0\}\}$.
    $G(\mathcal{A}) \leftarrow$ the digraph obtained from $G(\mathcal{A})$ by merging the vertices $\{\mathcal{I}_j, j \in J\}$ into the vertex $\mathcal{I}_{i_0}$.

Algorithm 4 ADD_CROSSING

update the archipelago $\mathcal{A}$ as follows:
    $E(\mathcal{A}) \leftarrow E(\mathcal{A}) \cup \{e\}$

Let us explain the algorithm. At the start, the archipelago $\mathcal{A}$ consists of the empty island with entry $\{x^*\}$. We then augment $\mathcal{A}$ one edge at a time, by adding firstly the edges of $\mathcal{A}$-type "new crossing" or "other" and then the edges of $\mathcal{A}$-type "crossing":

- Throughout the first While loop, $\mathcal{A}$ is an arborescent archipelago, as guaranteed by Propositions 4.6 and 4.9. It is very important to understand that, every time $\mathcal{A}$ is augmented in that loop, the vertices and entries of $\mathcal{A}$ may change, so the $\mathcal{A}$-types of the remaining edges may change as well: the $\mathcal{A}$-types of the edges in $E(\mathcal{H}) \setminus E(\mathcal{A})$ must be redetermined at each iteration of that loop.
- Throughout the second While loop, $\mathcal{A}$ is an archipelago, as guaranteed by Proposition 4.10. This time, the decomposition in islands does not change during that loop (we are adding crossing edges between already existing islands) so the $\mathcal{A}$-types of the remaining edges do not change.

That last remark proves that, after the two While loops, all remaining edges are of $\mathcal{A}$-type either "cut" or "exterior" (the $\mathcal{A}$-types "new crossing" and "other" have not reappeared during the second While loop). In conclusion, PARTITION_ARCHIPELAGO does output a partition of $E(\mathcal{H})$ and is therefore correct.

4.2.3 Time complexity

Let $n = |V(\mathcal{H})|$ and $m = |E(\mathcal{H})|$. We now show that PARTITION_ARCHIPELAGO runs in $O(m^2 k)$ time.

Let us first consider the three procedures ADD_NEWCROSSING, ADD_OTHER and ADD_CROSSING, to figure out how much time each update of $\mathcal{A}$ takes. Since basic operations on data structures can be language-dependent, let us clarify: when we use a list, what matters is the ability to remove the current element in $O(1)$ time; when we use an array, what matters is the ability to access and modify any element in $O(1)$ time.

- $E(\mathcal{H}) \setminus E(\mathcal{A})$ can be implemented as a list. Indeed, it is sensible to store $E(\mathcal{H}) \setminus E(\mathcal{A})$ rather than $E(\mathcal{A})$ since this is the set in which edges are searched for throughout. Each update consists in removing the current edge which is done in $O(1)$ time.
- $\mathcal{I}(\mathcal{A})$ can be implemented as an array of size n which contains, for each vertex $x \in V(\mathcal{H})$, the index of the island containing x (or 0 if $x \notin V(\mathcal{A})$). Each update requires going through the array once and is therefore done in $O(n)$ time.
- $\varepsilon(\mathcal{A})$ can be implemented as an array of size n which contains, for each vertex $x \in V(\mathcal{H})$, a 1 if x is in an entry of $\mathcal{A}$ or a 0 otherwise. Each update requires going through the array once and is therefore done in $O(n)$ time.
- $G(\mathcal{A})$ is an arborescence for the entire time that it is kept updated. Since $O(\frac{n}{k})$ islands are created in total (a new island can only be created during Add_NewCrossing, and this requires $k-1$ previously undiscovered vertices), $G(\mathcal{A})$ can be implemented as an array of size $O(\frac{n}{k})$ containing the parent of each island, *i.e.* for all index $i \neq 1$ it contains the only index j such that $(\mathcal{I}_j, \mathcal{I}_i) \in E(G(\mathcal{A}))$. In Add_NewCrossing, updating $G(\mathcal{A})$ is clearly done in $O(1)$ time. In Add_Other, updating $G(\mathcal{A})$ is done in $O(n)$ time: indeed, computing $|J_0| \leq k$ paths to the root takes $O(k \times \frac{n}{k}) = O(n)$ time, going through them a second time to compute i_0 and J takes $O(k \times \frac{n}{k}) = O(n)$ time again, and finally the merging process is performed in $O(\frac{n}{k})$ time since it only requires going through the array once.

All in all, performing Add_NewCrossing, Add_Other or Add_Crossing once is done in $O(n)$ time.

Determining the $\mathcal{A}$-type of a given edge e is easily done in $O(k)$ time since it boils down to determining, for all $x \in e$, which island/entry (if any) contains x.

We can now conclude on the time complexity of Partition_Archipelago:

- The initializations before the first While loop are done in $O(m+n)$ time.
- During the first While loop, finding an edge of $\mathcal{A}$-type "new crossing" or "other" and then adding it takes $O(mk+n)$ time: indeed, at most m edges are gone through (with the $\mathcal{A}$-type being determined for each one in $O(k)$ time as we have just seen) before finally finding one of $\mathcal{A}$-type "new crossing" or "other" which is added in $O(n)$ time as shown above. Since at most m edges of $\mathcal{A}$-type "new crossing" or "other" are added in total, the first While loop ends in $O(m(mk+n)) = O(m^2k + mn)$ time.
- During the second While loop, no $\mathcal{A}$-types need to be redetermined, and each update of $\mathcal{A}$ is done in $O(1)$ time so that this loop ends in $O(m)$ time.
- Finally, computing E_{cut} and E_{ext} at the very end of the algorithm takes $O(m)$ time.

In conclusion, Partition_Archipelago runs in $O(m^2k + mn)$ time. Since the $(k-2)$-linear connected component is a subset of the connected component, it is reasonable to assume that $\mathcal{H}$ is connected, which implies that $m \geq \frac{n-1}{k-1}$. Therefore, we can simplify $O(m^2k + mn)$ as $O(m^2k)$. This ends the proof of Theorem 4.4.

Notice that the algorithm can easily be tweaked so as to also return a $(k-2)$-linear path from x^* to x for each $x \in LCC_{\mathcal{H}}^{k-2}(x^*)$. Indeed, it suffices, throughout the algorithm, to keep in memory an (x^*, X)-extendable path in $\mathcal{A}$ for each $X \subset V(\mathcal{A})$ such that $1 \leq |X| \leq k-1$ and $X \notin \varepsilon(\mathcal{A})$, which is possible by following the construction given in the proof of Proposition 4.9. If $k = O(1)$ then the algorithm remains in polynomial time.

4.3 Proof of the main results

Proof of Theorem 4.2: Let $\mathcal{A}, E_{cut}, E_{ext}$ be as in Theorem 4.4.

- Let us first show that $\mathcal{A} = \mathcal{H}[LCC_{\mathcal{H}}^{k-2}(x^*)]$. Since no edge in $E_{cut} \cup E_{ext}$ is included in $V(\mathcal{A})$, we know $\mathcal{A}$ is an induced subhypergraph of $\mathcal{H}$. Moreover $V(\mathcal{A}) \subseteq LCC_{\mathcal{H}}^{k-2}(x^*)$ by Corollary 3.11, so it remains to verify that $LCC_{\mathcal{H}}^{k-2}(x^*) \subseteq V(\mathcal{A})$. The idea is simple: the only way to leave the archipelago is through an edge in E_{cut}, however a $(k-2)$-linear path in $\mathcal{A}$ from x^* to an entry of size $k-1$ necessarily contains that entry entirely, making it impossible to then use an edge in E_{cut} without violating the $(k-2)$-linearity. We now give the rigorous proof.
  Suppose for a contradiction that there exists $x \in LCC_{\mathcal{H}}^{k-2}(x^*) \setminus V(\mathcal{A})$. Let $\overrightarrow{P} = (e_1, \dots, e_L)$ be a $(k-2)$-linear path from x^* to x in $\mathcal{H}$. Since $x \notin V(\mathcal{A})$, we can define $M := \inf\{1 \leq p \leq L \text{ such that } e_p \not\subset V(\mathcal{A})\}$. Since all edges adjacent to x^* are necessarily in $E(\mathcal{A})$, we have $e_1 \subset V(\mathcal{A})$ hence $M \geq 2$. Moreover e_M intersects $e_{M-1} \subset V(\mathcal{A})$, so $e_M \in E_{cut}$ from which $e_M \cap V(\mathcal{A}) = \varepsilon$ for some entry ε of $\mathcal{A}$ of size $k-1$. Let $y \in e_M \cap e_{M-1} \subset \varepsilon$: since $\overrightarrow{Q} := (e_1, \dots, e_{M-1})$ is a $(k-2)$-linear path from x^* to y in $\mathcal{A}$, Proposition 3.8 ensures that $\varepsilon \subset e_{M-1}$. Since $\varepsilon \subset e_M$, this contradicts the $(k-2)$-linearity of $\overrightarrow{P}$.
- Any archipelago $\mathcal{A}'$ in $\mathcal{H}$ is a subhypergraph of $\mathcal{H}[LCC_{\mathcal{H}}^{k-2}(x^*)] = \mathcal{A}$, because $V(\mathcal{A}') \subseteq LCC_{\mathcal{H}}^{k-2}(x^*)$ by Corollary 3.11. This shows both that $\mathcal{A}$ is a maximal archipelago and that it is the only one.
- Finally, the complexity result is obvious since computing $\mathcal{H}[LCC_{\mathcal{H}}^{k-2}(x^*)] = \mathcal{A}$ is equivalent to computing $E(\mathcal{A})$. □

5 Consequences of the algorithmic result

5.1 Link with the Maker-Breaker positional game

A *positional game* is played on a hypergraph $\mathcal{H}$, with two players taking turns picking previously unpicked vertices of $\mathcal{H}$, and the winner is decided by one of several conventions. In the *Maker-Breaker* convention, the first player ("Maker") wins if she owns all vertices of some edge of $\mathcal{H}$, while the second player ("Breaker") wins if he can prevent this from happening. Note that, since both players have complementary goals, no draw is possible. The algorithmic problem consisting in deciding which player wins the Maker-Breaker game with optimal play is studied in the literature:

MakerBreaker
Input : a hypergraph $\mathcal{H}$.
Output : YES if and only if Maker wins the Maker-Breaker game on $\mathcal{H}$.

The MakerBreaker problem is trivially tractable on hypergraphs of rank 2 (Breaker wins on a graph G if and only if all connected components of G are of size 1 or 2), and is known to be PSPACE-complete on 6-uniform hypergraphs [Rahman and Watson (2021)]. In a separate paper

[Galliot et al. (2022)], we study the Maker-Breaker problem on hypergraphs of rank 3, in which linear paths play a crucial role. If $\mathcal{H}$ contains a linear path from x to y, where Maker owns x and y while the other vertices of the path are free (*xy-nunchaku*), then Maker easily wins when playing first, by forcing all of Breaker's moves along the path until Breaker is trapped. It is shown in [Galliot et al. (2022)] that Maker wins on a hypergraph of rank 3, when playing first, if and only if she has a strategy ensuring that the updated hypergraph contains a nunchaku at the end of one of the first four rounds of play. Therefore:

Theorem. [Galliot et al. (2022)] *MakerBreaker on hypergraphs of rank 3 reduces polynomially to* $\textsc{HypConnectivity}_{3,1}$.

Corollary 4.3 thus concludes that MakerBreaker is solvable in polynomial time on hypergraphs of rank 3. This validates part of a conjecture by [Rahman and Watson (2020)].

5.2 *Link with* PAFP

5.2.1 *Reducing* $\textsc{HypConnectivity}_{k,q}$ *to* PAFP

A first attempt at tackling the algorithmic complexity of $\textsc{HypConnectivity}_{k,q}$, for general $1 \le q \le k-2$, could be the following reduction to the "Paths Avoiding Forbidden Pairs" problem known as PAFP (sometimes PPFP or PFP):

PAFP	
Input	: a bicolored graph G (all edges are blue or red), and $x, y \in V(G)$.
Output	: YES if and only if there exists a blue induced path from x to y in G.

Notation 5.1. Let $\varphi_{k,q}$ be the function that associates to a k-uniform hypergraph $\mathcal{H}$ the bicolored graph G defined by:

- $V(G) = E(\mathcal{H})$;
- For all distinct $e_1, e_2 \in V(G)$, there is a blue (resp. red) edge between e_1 and e_2 in G if and only if $1 \le |e_1 \cap e_2| \le q$ (resp. if and only if $|e_1 \cap e_2| > q$).

Therefore G is simply the line graph of $\mathcal{H}$ with added colors that carry information on the size of the intersections. See Figure 18 for an example.

Proposition 5.2. *For all* $k \ge 3$ *and* $1 \le q \le k-2$*,* $\textsc{HypConnectivity}_{k,q}$ *polynomially reduces to PAFP.*

Proof: This is clear: by definition, a sequence of edges $(e_1, \dots, e_L)$ in $\mathcal{H}$ is a q-linear path if and only if it is a blue induced path in $\varphi_{k,q}(\mathcal{H})$ ("blue" means two consecutive edges intersect on between 1 and q vertices, "induced" means two non-consecutive edges do not intersect). Therefore, there exists a q-linear path from x to y in $\mathcal{H}$ ($x \neq y$) if and only if there exist edges $e_x \ni x$ and $e_y \ni y$ in $\mathcal{H}$ such that there exists a blue induced path between e_x and e_y in $\varphi_{k,q}(\mathcal{H})$. $\square$

However, PAFP is known to be NP-complete in general [Gabow et al. (1976)]. In fact, unless P=NP, there is no linear approximation ratio for the minimum number of red edges induced by a blue path between two given vertices [Hajiaghayi et al. (2012)]. For the problem on directed graphs (the blue edges are directed arcs), which is by far the most studied version in the literature, a few tractable cases are known but they are of little help to us:

- It is shown in [Yinnone (1997)] that the problem is tractable if the red edges form a matching and a *skew symmetry condition* is satisfied. Even though the undirected version is also true with basically the same proof, it does not solve HypConnectivity$_{k,q}$ since a general bicolored graph in $\mathrm{Im}(\varphi_{k,q})$ does not satisfy these conditions (nor does it easily reduce to one that does).
- Other tractable cases are addressed in [Chen et al. (2001)] and [Kolman and Pangrac (2009)], however they are very specific to directed acyclic graphs.

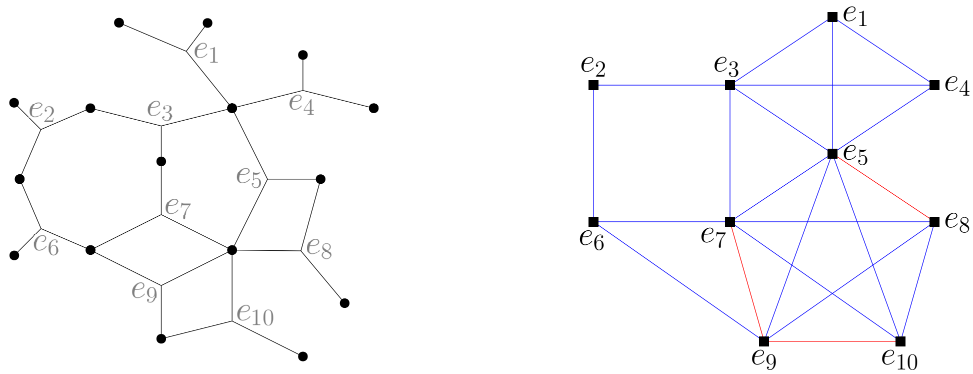

Fig. 18: On the left: a 3-uniform hypergraph $\mathcal{H}$. On the right: the bicolored graph $G = \varphi_{3,1}(\mathcal{H})$.

5.2.2 *Reducing some instances of* PAFP *to* HypConnectivity$_{k,q}$

Instead, now that we know HypConnectivity$_{k,k-2}$ is solvable in polynomial time for all $k \geq 3$, it is interesting to turn the tables and examine the implications on PAFP:

Theorem 5.3. *PAFP is tractable on bicolored graphs in* $\bigcup_{k\geq 3} \mathrm{Im}(\varphi_{k,k-2})$ *for which a preimage can be computed in polynomial time.*

Proof: Let $G = \varphi_{k,k-2}(\mathcal{H})$ for some k-uniform hypergraph $\mathcal{H}$, and let $e, e' \in V(G) = E(\mathcal{H})$ be distinct. As we have seen before, the blue induced paths between e and e' in G are exactly the $(k-2)$-linear paths $(e = e_1, \ldots, e_L = e')$ in $\mathcal{H}$. Since HypConnectivity$_{k,k-2}$ requires a start vertex and an end vertex in its input, define, for all $x \in e$ and $y \in e'$, the hypergraph $\mathcal{H}_{x,y}$ obtained from $\mathcal{H}$ by removing all edges adjacent to x and y other than e and e', so that any $(k-2)$-linear path from x to y in $\mathcal{H}_{x,y}$ necessarily starts with e and ends with e'. There exists a blue induced path between e and e' in G if and only if there exist $x \in e$ and $y \in e'$ such that there is a $(k-2)$-linear path from x to y in $\mathcal{H}_{x,y}$, which concludes since HypConnectivity$_{k,k-2}$ is solvable in polynomial time. $\square$

Therefore, any sufficient condition for a bicolored graph G to be in $\text{Im}(\varphi_{k,k-2})$ for some $k \geq 3$, if it can be checked in polynomial time and comes with a way to reconstruct a preimage hypergraph in polynomial time, would add to the very short list of known tractable cases for PAFP.

For standard (*i.e.* non-colored) line graphs, the recognition problem has been studied extensively. Line graphs of graphs are characterized by a finite list of forbidden induced subgraphs ("FIS") [Beineke (1970)]. Line graphs of hypergraphs, on the other hand, are notoriously difficult to recognize. There is no finite FIS characterization for line graphs of k-uniform hypergraphs if $k \geq 3$ [Lovász (1977)], and this recognition problem is even known to be NP-complete for $k = 3$ [Poljak et al. (1981)]. However, adding information about the size of the pairwise intersections of (hyper)edges, instead of simply telling which ones are non-empty, changes the problem. For example, if all these sizes are given and in $\{0, 1\}$ (which is equivalent to asking the hypergraph to be *linear*) then, while remaining NP-complete for $k = 3$ [Poljak et al. (1981)] [Hlineny and Kratochvil (1997)], the problem becomes easier in some cases:

- For $k = 3$, there is a finite FIS characterization for line graphs of 3-uniform linear hypergraphs if the minimum vertex-degree of the graph is at least 69, as well as a polynomial time algorithm to reconstruct the hypergraph in the positive case [Naik et al. (1982)]. This bound has since been improved from 69 to 16 for the finite FIS characterization and 10 for the tractability of the recognition problem [Skums et al. (2009)]. There is no analogous result for $k \geq 4$, no matter what constant lower bound is put on the minimum vertex-degree [Metelsky and Tyshkevich (1997)].
- For any $k \geq 3$, there is a finite FIS characterization for line graphs of k-uniform linear hypergraphs if the minimum edge-degree of the graph is at least $f(k)$, where f is a polynomial function, as well as a polynomial (whose power increases with k) time algorithm to reconstruct the hypergraph in the positive case [Naik et al. (1982)]. This result has been generalized by replacing the linearity of the hypergraph by any constant upper bound on its *multiplicity* [Bhattacharya et al. (2021)].

These results bring some hope of a finite FIS characterization for bicolored line graphs under some similar restriction over the minimum vertex-degree or edge-degree of the graph, and of a way to reconstruct a preimage in polynomial time which we crucially need. The case $k = 3$ is the most promising because the exact size of each intersection is also given (in $\{0, 1, 2\}$: $0 =$ no edge, $1 =$ blue edge, $2 =$ red edge), although it is NP-complete in general since instances with all blue edges correspond to the 3-uniform linear case for standard line graphs which we know is NP-complete. Figure 19 features some induced bicolored subgraphs that cannot appear in a bicolored graph from $\bigcup_{k\geq 3} \text{Im}(\varphi_{k,k-2})$. For instance, an induced red path on three vertices is impossible because, in a k-uniform hypergraph with $k \geq 3$, if $|e_1 \cap e_2| = |e_2 \cap e_3| = k - 1$ then $|e_1 \cap e_3| \geq k - 2 > 0$.

6 Conclusion and perspectives

In this paper, we have introduced q-linear paths in hypergraphs of rank k, and in the case $q = k - 2$ we have described the structure of the $(k-2)$-linear connected components as well as a polynomial time algorithm to compute them. The time complexity in $O(m^2 k)$ might be optimal,

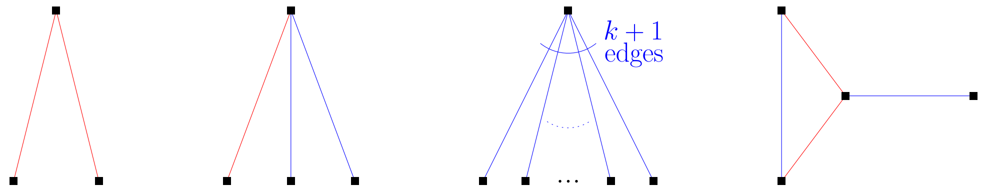

Fig. 19: Some induced subgraphs that cannot appear in $G \in \mathrm{Im}(\varphi_{k,k-2})$.

since it seems difficult to avoid an "accept or put aside" process on the edges where each edge is potentially examined $O(m)$ times, and the mere computation of the intersection of two edges is in $O(k)$ time.

What about other values of q? The linear case $q = 1$ is of particular interest, since linear paths appear in numerous other problems. However, if we want to try and generalize our techniques while maintaining a time complexity that is polynomial in k, it might be more reasonable to look at the case $q = k - c$ where $c \geq 3$ is a constant, with adapted definitions of islands and archipelagos (whose entries would be of size between $k - c + 1$ and $k - 1$). As an illustration of the difficulties that can be encountered during the algorithm, consider the case $k = 4$ and $q = 1$, where at some point an edge $e = \{x, y, z, t\}$ is discovered with x, y already known vertices from different islands and z, t unknown: on one hand e could be part of a new merged island (since $x, y \in e$), but on the other hand e could be a crossing edge towards a new island with entry $\{z, t\}$ (since z and t are not separated), and it seems hard to conciliate the two.

The bicolored line graph recognition problem is open. As mentioned in Section 5, the added information on the size of the pairwise intersections of edges might make this problem somewhat easier compared to standard line graphs, especially in the case $k = 3$. The characterization of line graphs of hypergraphs by a Krausz partition into cliques [Naik et al. (1982)] is easily adaptable to the bicolored version. Some characterizations by finite families of induced subgraphs from [Naik et al. (1982)] and their proofs might be adaptable as well, which would yield new classes of tractable instances for PAFP. Looking beyond applications to PAFP, a general *weighted line graph* recognition problem, where each edge of the graph would wear a number between 1 and $k - 1$ indicating the exact size of the corresponding intersection, seems interesting in itself.

Acknowledgements

We would like to thank the anonymous reviewers for their suggestions and comments.

References

X. Allamigeon. On the complexity of strongly connected components in directed hypergraphs. *Algorithmica*, 69(2):335–369, 2014. doi: 10.1007/s00453-012-9729-0.

L. W. Beineke. Characterizations of derived graphs. *J. Combin. Theory*, 9(2):129–135, 1970. doi: 10.1016/S0021-9800(70)80019-9.

A. Bhattacharya, A. Godinho, P. Majumder, and N. M. Singhi. Reconstruction of hypergraphs from line graphs and degree sequences. Preprint: `https://arxiv.org/abs/2104.14863`, 2021.

T. Chen, M.-Y. Kao, M. Tepel, J. Rush, and G. M. Church. A dynamic programming approach to de novo peptide sequencing via tandem mass spectrometry. *J. Comput. Biol.*, 8(3):325—337, 2001. doi: 10.1089/10665270152530872.

A. Dudek, S. La Fleur, D. Mubayi, and V. Rödl. On the size-Ramsey number of hypergraphs. *J. Graph Theory*, 86(1):104–121, 2017. doi: 10.1002/jgt.22115.

Z. Füredi, T. Jiang, and R. Seiver. Exact solution of the hypergraph Turán problem for k-uniform linear paths. *Combinatorica*, 34(3):299–322, 2014. doi: 10.1007/s00493-014-2838-4.

H. N. Gabow, S. N. Maheswari, and L. J. Osterweil. On two problems in the generation of program test paths. *IEEE Trans. Softw. Eng.*, 2(3):227–231, 1976. doi: 10.1109/tse.1976.233819.

F. Galliot, S. Gravier, and I. Sivignon. Maker-Breaker is solved in polynomial time on hypergraphs of rank 3. Preprint: `https://arxiv.org/abs/2209.12819`, 2022.

G. Gallo, G. Longo, and S. Pallottino. Directed hypergraphs and applications. *Discrete Appl. Math.*, 42(2-3):177–201, 1993. doi: 10.1016/0166-218X(93)90045-P.

R. Gu, J. Li, and Y. Shi. Anti-Ramsey numbers of paths and cycles in hypergraphs. *SIAM J. Discrete Math.*, 34(1):271–307, 2020. doi: 10.1137/19M1244950.

A. Guzzo, A. Pugliese, A. Rullo, and D. Saccà. Intrusion detection with hypergraph-based attack models. In *Graph Structures for Knowledge Representation and Reasoning*, proc. Third International Workshop, GKR 2013, Beijing, China, volume 8323 of *LNCS*, pages 58–73, Cham, 2014. Springer International Publishing. doi: 10.1007/978-3-319-04534-4_5.

M. T. Hajiaghayi, R. Khandekar, G. Kortsarz, and J. Mestre. The checkpoint problem. *Theoret. Comput. Sci.*, 452:88–99, 2012. doi: 10.1016/j.tcs.2012.05.021.

P. Hlineny and J. Kratochvil. Computational complexity of the Krausz dimension of graphs. In *Graph-Theoretic Concepts in Computer Science*, proc. 23rd International Workshop, WG'97, Berlin, Germany, volume 1335 of *LNCS*, pages 214–228, Berlin, 1997. Springer. doi: https://doi.org/10.1007/BFb0024500.

E. Jackowska. The 3-color Ramsey number for a 3-uniform loose path of length 3. *Australas. J. Combin.*, 63(2):314–320, 2015.

E. Jackowska, J. Polcyn, and A. Ruciński. Turán numbers for 3-uniform linear paths of length 3. *Electron. J. Combin.*, 23(2):P2.30, 2016. doi: 10.37236/5320.

P. Kolman and O. Pangrac. On the complexity of paths avoiding forbidden pairs. *Discrete Appl. Math.*, 157(13):2871–2876, 2009. doi: 10.1016/j.dam.2009.03.018.

L. Lovász. Problem 9. In *Beiträge zur Graphentheorie und deren Anwendungen*, proc. Internationalen Kolloquium, Oberhof, DDR, page 313, 1977.

Y. Metelsky and R. I. Tyshkevich. On line graphs of linear 3-uniform hypergraphs. *J. Graph Theory*, 25(4):243–251, 1997. doi: 10.1002/(SICI)1097-0118(199708)25:4<243::AID-JGT1>3.0.CO;2-K.

R. N. Naik, S. B. Rao, S. S. Shrikhande, and N. M. Singhi. Intersection graphs of k-uniform linear hypergraphs. *European J. Combin.*, 3:159–172, 1982. doi: 10.1016/S0195-6698(82)80029-2.

G. R. Omidi and M. Shahsiah. Ramsey numbers of 3-uniform loose paths and loose cycles. *J. Combin. Theory Ser. A*, 121:64–73, 2014. doi: 10.1016/j.jcta.2013.09.003.

S. Poljak, V. Rödl, and D. Turzík. Complexity of representation of graphs by set systems. *Discrete Appl. Math.*, 3(4):301–312, 1981. doi: 10.1016/0166-218X(81)90007-X.

M. L. Rahman and T. Watson. Tractable unordered 3-CNF games. In *LATIN 2020: Theoretical Informatics*, proc. Latin American Symposium on Theoretical Informatics, volume 12118 of *Lecture Notes in Comput. Sci.*, pages 360–372, Cham, 2020. Springer International Publishing. doi: 10.1007/978-3-030-61792-9_29.

M. L. Rahman and T. Watson. 6-uniform Maker-Breaker game is PSPACE-complete. In *38th International Symposium on Theoretical Aspects of Computer Science (STACS 2021)*, volume 187 of *LIPIcs*, pages 57:1–57:15, Dagstuhl, Germany, 2021. Schloss Dagstuhl, Leibniz-Zentrum für Informatik. doi: 10.4230/LIPIcs.STACS.2021.57.

P. V. Skums, S. V. Suzdal, and R. I. Tyshkevich. Edge intersection graphs of linear 3-uniform hypergraphs. *Discrete Math.*, 309(11):3500–3517, 2009. doi: 10.1016/j.disc.2007.12.082.

M. Thakur and R. Tripathi. Linear connectivity problems in directed hypergraphs. *Theoret. Comput. Sci.*, 410(27-29):2592–2618, 2009. doi: 10.1016/j.tcs.2009.02.038.

I. Tomescu. Some results on chromaticity of quasi-linear paths and cycles. *Electron. J. Combin.*, 19(2):P23, 2012. doi: 10.37236/2370.

B. Wu and Y. Peng. Lagrangian densities of short 3-uniform linear paths and Turán numbers of their extensions. *Graphs Combin.*, 37(3):711–729, 2021. doi: 10.1007/s00373-020-02270-w.

H. Yinnone. On paths avoiding forbidden pairs of vertices in a graph. *Discrete Appl. Math.*, 74 (1):85–92, 1997. doi: 10.1016/S0166-218X(96)00017-0.