Partitioning Vertices and Edges of Graphs into Connected Subgraphs

Olivier Baudon, Julien Bensmail, Lyn Vayssieres

Partitioning Vertices and Edges of Graphs into Connected Subgraphs

Olivier Baudon[b], Julien Bensmail[a], Lyn Vayssieres[b]

[a] *Université Côte d'Azur, CNRS, Inria, I3S, France*
[b] *Univ. Bordeaux, CNRS, Bordeaux INP, LaBRI, UMR 5800, F-33400, Talence, France*

Abstract

Arbitrarily partitionable (AP) graphs are graphs that can have their vertices partitioned into arbitrarily many parts inducing connected graphs of arbitrary orders. Since their introduction, several aspects of AP graphs have been investigated in literature, including structural and algorithmic aspects, their connections with other fundamental notions of graph theory, and variants of the original notion. Quite recently, an edge version of AP graphs, called arbitrarily edge-partitionable (AEP) graphs have been introduced and studied, with a special focus on their similarities and discrepancies with AP graphs.

In this work, we introduce and study a total variant of AP graphs, called arbitrarily total-partitionable (ATP) graphs, which essentially stand as a combination of AP and AEP graphs, for some particular notion of connectivity for sets of vertices and edges. We establish results of several natures, which we compare to known, similar results for AP and AEP graphs. In particular, we prove that, although the involved definitions are rather close, being AP, AEP, and/or ATP for a graph does not guarantee it also has the other properties. We also establish that deciding whether a tree can be partitioned in this total way is NP-complete in general, and provide sufficient conditions for ATPness in terms of longest paths. We finally raise directions for further work on the topic.

Keywords: arbitrarily partitionable graph; partition into connected graphs; total graph.

1. Introduction

In this work, we deal with some problems where one aims at **partitioning** some **elements of some graph** so that a **certain number** of **connected subgraphs** with **certain numbers of elements** result, for some notions of connectivity. More precisely, with deal with three distinct such problems, where the elements to be partitioned are **vertices**, **edges**, and **both**, respectively. In particular, the latter of these three problems is a new one we introduce, generalising the former two. So that our motivations and definitions are clear, we thus start by recalling what these two former problems are about.

Let $n, m \geq 1$ be two integers. To make upcoming definitions more uniform, for convenience we define an (n, m)-*graph* as a graph of order (number of vertices) and size (number of edges) n and m, respectively. For an integer $x \geq 1$, an *x-partition* $\pi = (\lambda_1, \ldots, \lambda_p)$ is a partition of x, that is, $\lambda_1 + \cdots + \lambda_p = x$. Let now G be an (n, m)-graph, and $\pi = (\lambda_1, \ldots, \lambda_p)$ be an n-partition. A *vertex-realisation* of π in G is a partition $(S_1, \ldots, S_p)$ of $V(G)$ such that $G[S_i]$ (the subgraph of G induced by S_i) is a connected graph of order λ_i, for all $i \in \{1, \ldots, p\}$. Now, we say G is *arbitrarily partitionable* (AP, for short) if all n-partitions are vertex-realisable in G; that is, we say G is AP if G can be partitioned into arbitrarily many connected graphs of arbitrary, requested orders.

AP graphs were introduced independently by Barth, Baudon, and Puech in [1], and by Horňák and Woźniak in [14]. In particular, the former authors introduced this notion to

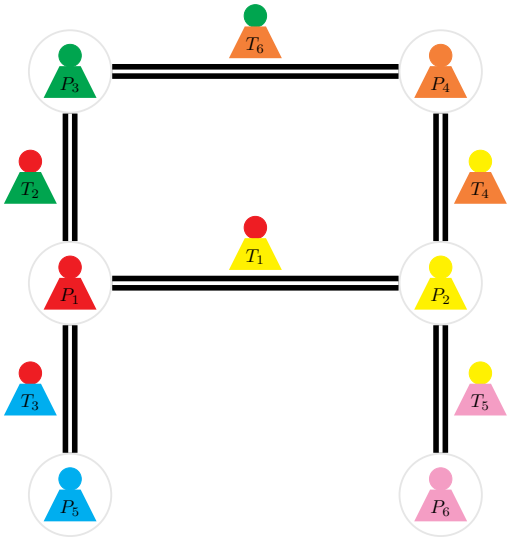
model a practical network sharing problem. To date, many aspects of APness have been investigated in literature, leading to results of different natures, showing that APness relates to several other graph theoretical notions. For instance, vertex-realisations of partitions containing occurrences of value 2 are directly equivalent to finding matchings fulfilling some properties. Also, APness is closed under adding edges, from which it can be observed that any traceable graph (*i.e.*, having a Hamiltonian path) is AP. For the sake of keeping the current introduction short, we will not elaborate any more here on what is known on AP graphs; we refer the reader to the most recent works on the topic for more information (see *e.g.* [4, 5, 8, 15, 16, 21]). Also, be aware that facts on AP graphs will be reminded throughout this work, as they are needed to catch the importance of our results.

More recently, Bensmail introduced in [7] an edge version of AP graphs, based on the following definitions. For an (n, m)-graph G and an m-partition $\pi = (\lambda_1, \ldots, \lambda_p)$, an *edge-realisation* of π in G is a partition $(S_1, \ldots, S_p)$ of $E(G)$ such that $G[S_i]$ (the subgraph of G obtained by keeping the edges in S_i only) is a connected graph of size λ_i, for all $i \in \{1, \ldots, p\}$. In turn, we say G is *arbitrarily edge-partitionable* (AEP, for short) if it turns out that all m-partitions are indeed edge-realisable in G.

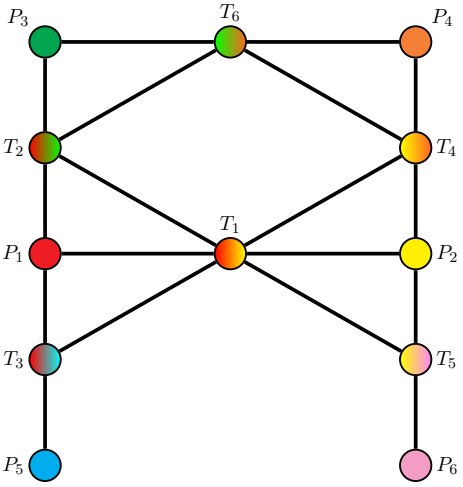
Although the main intent in [7] was to investigate how generalising APness to edges behaves, the author soon noticed that AEPness of graphs is nothing but APness of line graphs (graphs of adjacent edges). For this reason, quite expectedly, several results on APness were derived to AEPness in [7], including structural and algorithmic ones. Again, these details will be given later, as they connect to our investigations in the current work.

As mentioned earlier, our main goal in this work is to introduce and study a total variant of the earlier two problems, somewhat generalising them. As seen in the previous notions, a crucial point is the notion of connectivity considered for some sets of graph elements; since such a notion is not that obvious when considering sets containing both vertices and edges, we first need to explicit the notion of connectivity we adopt. Let G be an (n, m)-graph. We say two elements (vertices and/or edges) a and b of G are *touching* if, and only if, a and b are a vertex and one of its incident edges, or two adjacent edges (having a vertex in common). In particular, if a and b are two vertices, then they are never considered to be touching (even if a and b are adjacent). Now, assuming $S \subseteq V(G) \cup E(G)$ is a set of vertices and edges of G, we say S is *connected* if, for any two elements $a, b \in S$, there exist $x \geq 0$ other elements $c_1, \ldots, c_x$ of S such that all of the pairs $\{a, c_1\}, \{c_1, c_2\}, \ldots, \{c_{x-1}, c_x\}, \{c_x, b\}$ contain two touching elements each. Now, for an $(n+m)$-partition $\pi = (\lambda_1, \ldots, \lambda_p)$, a *total-realisation* of π in G is a partition $(S_1, \ldots, S_p)$ of $V(G) \cup E(G)$ such that S_i is connected and $|S_i| = \lambda_i$, for all $i \in \{1, \ldots, p\}$. Finally, we say G is *arbitrarily total-partitionable* (ATP, for short) if all $(n + m)$-partitions admit total-realisations in G.

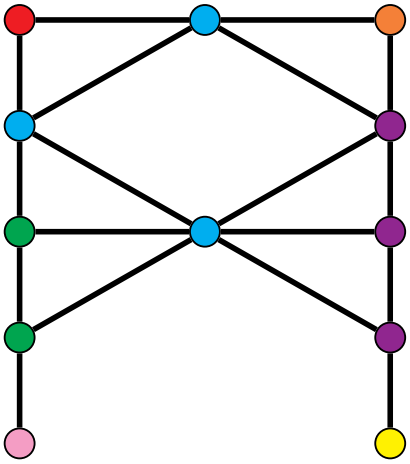
We have several sources of motivation for defining ATP graphs the way we do, and, in particular, for focusing on the notion of connectivity we have just introduced. To begin with, let us remind that AP graphs were introduced to deal with a practical network sharing problem [1]. Similarly, our notion of ATP graphs can be used to model a particular problem, defined as follows (see Figure 1 for an illustration), which is not too artificial. Suppose that a particular event occurs, gathering n persons $P_1, \ldots, P_n$, each speaking their own language only (that is, to be clear, no two persons speak the same language). So that some of these persons can communicate, a number of translators $T_1, \ldots, T_m$ are hired, where each translator T_i speaks exactly two of the spoken languages, and can thus allow exactly two of the P_i's to communicate. Assume now that the organisers wish to group the attendees into p groups containing $\lambda_1, \ldots, \lambda_p$ people (persons and/or translators) each, such that, within each group, any two people can communicate, either directly because they speak the same language, or through intermediate translators. Then designing such groups is equivalent



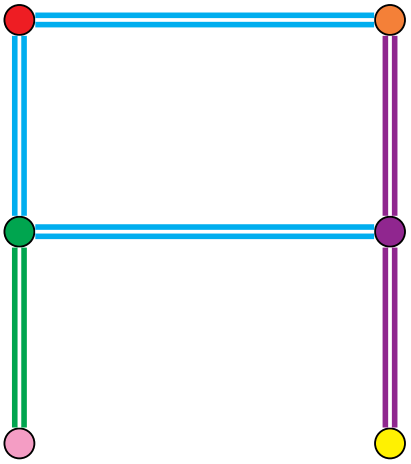
(a) Original graph G



(b) Total graph $T(G)$



(c) V.-real. of $(1,1,1,1,2,3,3)$ in $T(G)$



(d) T.-real. of $(1,1,1,1,2,3,3)$ in G

Figure 1: Example of situation for the practical problem described in the introductory section. We here have six persons $P_1,\ldots,P_6$ and six translators $T_1,\ldots,T_6$, where $T_1,\ldots,T_6$ allow for the pairs $\{P_1,P_2\},\{P_1,P_3\},\{P_1,P_5\},\{P_2,P_4\},\{P_2,P_6\},\{P_3,P_4\}$ to communicate (a). (b) represents the pairs of persons and translators who speak a common language. (c) shows a grouping of these twelve people into seven groups with size in $(1,1,1,1,2,3,3)$ able to communicate (not necessarily directly). (d) shows these groups in the original graph. In (c) and (d), groups are represented as elements having a same colour.

to finding a total-realisation of the $(n+m)$-partition $(\lambda_1,\ldots,\lambda_p)$ in the (n,m)-graph G having a vertex v_i corresponding to each person P_i, and an edge v_iv_j for any translator allowing (only) for P_i and P_j to communicate. In particular, note that, within a group, two people able to communicate directly are either a person and a translator speaking their language, or two translators speaking some same language (thus being translators for some same person). Meanwhile, by definition, in any group any two persons P_i and P_j cannot communicate directly: they need a translator. W.r.t. those concerns, it would make more sense that the hired translators allow for the p groups above to be created whatever the numbers $\lambda_1,\ldots,\lambda_p$ of people we want each group to contain, and whatever the number p of groups is. That is, the graph G modelling the configuration should be ATP.

Still regarding why we define things the way we do, we believe the way we define

connectivity in our context has a flavour of definitions behind both APness and AEPness. In particular, our notion of connectivity relies on the notion of touching elements. Parts in total-realisations are connected either due to adjacent edges, or due to vertices and incident edges. The fact that the former pairs of touching elements contribute to connectivity is directly inspired by AEPness, since, in edge-realisations, the fact that a part is connected relies solely on the fact that it includes pairs of adjacent edges. The fact that the latter pairs contribute to connectivity is inspired from APness, since, in vertex-realisations, the fact that a part is connected does not rely solely on the fact that it contains adjacent vertices, but rather on the fact that these adjacent vertices are joined by edges. There is a catch, however, being that, in vertex-realisations, the numbers of edges in connected parts do not take part to the numbers of elements that we require parts to contain. Still, for neither APness nor AEPess, having two non-adjacent vertices in some part of some realisation does not contribute to connectivity of said part, which is why, in our definition of touching elements, we exclude pairs of vertices (regardless of whether they are adjacent).

Our main intent in this work is to give a first insight into ATPness. In particular, an interesting point in our opinion is comparing the behaviours of ATPness and those of APness and AEPness. We do it through several successive sections. We start in Section 2 by raising first remarks on ATPness, and, in particular, by making more clear obvious relationships between ATPness and both APness and AEPness. Then, in Section 3, we prove that, although the definition of ATP graphs is directly inspired by the definitions of AP and AEP graphs, being both AP and AEP for a graph does not guarantee it is necessarily ATP, thereby legimating our new problem. In that section, we also establish other results of this type, showing in particular that there are arbitrarily large graphs that are AP, AEP, or ATP, but do not have the other properties. In Section 4, we then focus on complexity aspects, showing mainly that determining whether some partition admits a total-realisation in some graph is NP-complete in general. A remarkable fact is that we show such a result to hold for very simple graph structures, namely for trees. Last, we establish sufficient conditions for ATPness in Section 5, where said conditions are in terms of longest paths and are inspired by similar ones for APness and AEPness. We finish off in Section 6 with concluding remarks, including directions for further work on the topic.

2. First remarks on ATP graphs

Let us begin by making more clear the equivalences between APness, AEPness, and ATPness. First off, as mentioned earlier, and as observed in [7], it should be clear that finding an edge-realisation of some partition π in some graph G is equivalent to finding a vertex-realisation of π in $L(G)$, the *line graph* of G (*i.e.*, having a vertex v_e for each edge $e \in E(G)$, and an edge $v_e v_f$ whenever two edges e and f share a vertex in G). Likewise, since, in total-realisations, the notion of connectivity relies solely on the notion of touching elements we introduced earlier, we can define a graph transformation through which we can translate total-realisations into vertex-realisations. Namely, we define the *total graph* $T(G)$ of G as the graph of its touching elements. That is, $T(G)$ has a vertex v_x for every element $x \in V(G) \cup E(G)$, and an edge $v_x v_y$ for every two touching elements x and y.

Observation 2.1. *Any total-realisation of some partition π in a graph G yields a vertex-realisation of π in $T(G)$, and vice versa. Thus, G is ATP if and only if $T(G)$ is AP.*

Thus, studying ATPness of graphs is equivalent to studying APness of particular graphs, namely graphs being total graphs of some graphs. Looking at the structure of total graphs, we note that, by definition, for any graph G, the line graph $L(G)$ of G is strictly included

in $T(G)$. More precisely, $T(G)$ is obtained from $L(G)$ by considering every vertex x of G, and adding a new vertex v_x and all edges $v_x v_e$ such that e is an edge of G incident to x.

Although this is a bit less natural, let us mention that we could also establish that finding total-realisations in some graph G is equivalent to finding edge-realisations in a graph constructed from G. Namely, it can be observed that finding a total-realisation of any partition π in G is equivalent to finding an edge-realisation of π in $G \odot K_1$, the *corona product* of G and K_1 (obtained from G through attaching a pendant edge at every vertex). Note in particular that $L(G \odot K_1)$ is nothing but $T(G)$.

As mentioned earlier, AP graphs are quite related to matchings. In particular, it can be observed that finding a vertex-realisation of $(2, \ldots, 2)$ in a graph of even order is equivalent to finding a perfect matching. On the edge side of things, finding an edge-realisation of $(2, \ldots, 2)$ in a graph of even size is equivalent to finding a decomposition (*i.e.*, a partition of the edge set) into paths of length 2, which is always possible (as recalled in [7]). On this aspect, ATP graphs are closer to AEP graphs, in the sense that graphs with an even number of elements all admit total-realisations of $(2, \ldots, 2)$.

Observation 2.2. *Every connected (n, m)-graph with $n+m$ even admits a total-realisation of $(2, \ldots, 2)$.*

Proof. We prove a slightly stronger claim; namely, we prove the claim to hold for connected sets of vertices and edges. That is, for structures in which we may have adjacent edges not sharing a vertex, and edges incident to one or no vertex. In some sense, we thus focus on sets for which the corresponding total graph is connected. For this proof only, for convenience we still call (n, m)-graph such a set containing n vertices and m edges.

We prove the stronger claim by induction on $n+m$, for an (n, m)-graph G (with $n+m \geq 2$ even). The base case is when $n+m = 2$, which corresponds to when G is either a vertex and an incident edge (thus incident to one vertex only), or two adjacent edges (with no incident vertices). In these cases, clearly the whole set of elements of G forms a total-realisation of (2), and we are done. So, from now on, let us focus on the general case.

To begin with, all edges of G cannot be bridges. Indeed, this would imply G is a tree, and, here, we could proceed as follows. If G has a vertex u of degree 1, thus incident to a single edge e, then we can consider $\{u, e\}$ as a connected part of size 2, and apply induction on the $(n-1, m-1)$-graph $G - \{u, e\}$ to deduce the rest of a total-realisation of $(2, \ldots, 2)$. Otherwise, if G does not have a vertex of degree 1, then we can root G at any pendant edge r, and consider a pendant edge e at maximum (deepest) distance d from r. By the choice of e, either e is adjacent to an edge f at the same distance d from r (and then we consider $\{e, f\}$ and apply induction on $G - \{e, f\}$, which is connected since there are no degree-1 vertices in G), or e is adjacent to an edge f at distance $d-1$ from r. In the latter case, either e and f share a vertex u, in which case we consider $\{e, u\}$ and apply induction in $G - \{e, u\}$, or they do not, in which case we consider $\{e, f\}$ and apply induction in $G - \{e, f\}$. Both cases, note that the rest of the graph indeed remains connected.

So, we can now assume that G has an edge e_0 that is not a bridge. Then e_0 belongs to a cycle C_0. To make things clearer, we restrict C_0 to its edges only. If e_0 is incident to a vertex u, then note that $G - \{u, e_0\}$ remains connected, and so we are done as earlier. We have the same conclusion if e_0 is adjacent to a pendant edge (not on C_0). So, now, assume e_0 is incident to no vertex and adjacent to no pendant edge. Let e'_0 denote any of the two edges of C_0 adjacent to e_0. If $G - \{e_0, e'_0\}$ is connected, then, again, we are done. Otherwise, this means e_0 and e'_0 are adjacent to a bridge f_1. If the connected component G_1 of $G - \{e_0, e'_0\}$ containing f_1 is a tree (thus containing at least two elements, since f_1 is a bridge), then note that we could be done as in the case of trees earlier (setting f_1

as the root). Thus, G_1 contains a cycle $C_1 \neq C_0$ (of edges only), to which f_1 does not belong. By repeating, to C_1, the exact same arguments as for C_0, either we come up with a total-realisation of $(2, \dots, 2)$ in G, or we deduce that the edges of C_1 are incident to no vertex and adjacent to no pendant edge, and that any two of its adjacent edges are adjacent to bridges. In particular, we can find a bridge f_2 adjacent to two adjacent edges e_1, e_1' of C_1 such that the connected component G_2 of $G - \{e_1, e_1'\}$ containing f_2 contains a cycle $C_2 \neq C_0, C_1$. And then the previous arguments repeat.

At any point, either earlier arguments apply to deduce a total-realisation of $(2, \dots, 2)$ in G, or we deduce a collection $f_1, f_2, \dots$ of bridges, such that each f_i is adjacent to two adjacent edges e_{i-1} and e_{i-1}' (on a cycle C_{i-1}) where the connected component G_i of $G - \{e_{i-1}, e_{i-1}'\}$ containing f_i contains a cycle C_i (of edges) different from $C_0, \dots, C_{i-1}$. In particular, note that $|V(G_i)| + |E(G_i)| > |V(G_{i+1})| + |E(G_{i+1})|$ for all $i \geq 1$. Since G has a finite number of elements, the process must stop with a last bridge f_k, and previous arguments must apply around C_k. Thus, we can apply induction, and the claim holds. □

Another crucial aspect of APness, is that this property is closed under adding edges. Phrased differently, any graph spanned by an AP graph is AP itself. As established in [7], this does not hold true for AEPness, in the sense that there are AEP graphs which, upon being repeatedly added edges, repeatedly loose and regain the AEP property at certain points. We prove the same phenomenon can occur for ATPness as well, thereby showing that, regarding this aspect, ATPness is closer to AEPness than to APness.

Observation 2.3. *There are non-ATP graphs that are spanned by ATP graphs.*

Proof. Consider *e.g.* the following construction. Let $d \geq 1$ be any positive integer, and consider G the graph obtained as follows:

- start from a single vertex r;
- add three vertices a, b, c to G, and make them all adjacent to r only;
- add to G a path $d_1 \dots d_d$ on d vertices, and add the edge rd_1.

In other words, G is obtained from a star $K_{1,4}$ with four leaves by subdividing one edge exactly $d-1$ times. Then G is an (n, m)-graph for $n = d + 4$ and $m = d + 3$. Now, let π be any $(n + m)$-partition; we claim π admits a total-realisation in G whenever it contains an element different from 3.

- If $1 \in \pi$, then pick $\{a\}$ as a connected part. It is not too complicated to check that the rest of π admits a total-realisation in the rest of the graph (see mostly *e.g.* later Lemma 5.2 for more details, if this is unclear).
- If $2 \in \pi$, then pick $\{a, ar\}$. We then have a similar conclusion.
- If $4 \in \pi$, then pick $\{a, ar, b, br\}$. The rest of the graph is then a path, in which we can easily deduce the rest of the total-realisation
- If $5 \in \pi$, then pick $\{a, ar, b, br, r\}$. We then reach a similar conclusion.
- If $\lambda \in \pi$ for some $\lambda \geq 6$, then pick the part containing a, ar, b, br, c, and cr, and, from here, add the $\lambda - 6$ first elements of $(r, rd_1, d_1, d_1d_2, d_2, \dots)$. Then what remains is essentially a path, in which the rest of the total-realisation can easily be deduced.

Thus, only partitions π of the form $(3, \ldots, 3)$ can potentially be problematic. Actually, we note that a total-realisation of such a π would have to contain a part S of the form $\{a, ar, r\}$, $\{a, ar, rb\}$ (or $\{a, ar, rc\}$), or $\{a, ar, rd_1\}$. The latter two situations are not valid, however, as $G - S$ would contain a connected component containing one or five elements. Thus, such a total-realisation would have to contain $\{a, ar, r\}$, but then one of the other parts would have to be $\{b, br, rc\}$ or $\{b, br, rd_1\}$, leading to a similar conclusion. In other words, if $n + m \equiv 0 \bmod 3$, then G is not ATP (while G is ATP in all other cases).

To see now the claim holds true, just observe that adding any number of edges among the d_i's has essentially no effect over the realisability properties above (such edges can easily be added to parts; see, again, the proof of later Lemma 5.2 if this is unclear). In particular, whenever adding a number of edges such that, still calling m the resulting number of edges, we have $n + m \not\equiv 0 \bmod 3$, the resulting graph is ATP. Otherwise, it is not. We thus get our conclusion from repeatedly adding edges this way to G (and having d large enough). □

An interesting consequence of the fact that every graph spanned by an AP graph is AP itself, is that traceable graphs are all AP (since paths are obviously AP). In [7], Bensmail observed that, w.r.t. AEPness, a similar property is that all edge-traceable graphs (*i.e.*, graphs having a edge-Hamiltonian path, which is an ordering of the edges such that every two consecutive edges are adjacent; essentially, this corresponds exactly to a Hamiltonian path in the line graph) are AEP. We could as well define a *total-Hamiltonian path* of some graph G as an ordering over the elements (vertices and edges) of G in which every two consecutive elements are touching (again, this corresponds exactly to a Hamiltonian path in the total graph $T(G)$), and say G is *total-traceable* if it contains a total-Hamiltonian path. Then, obviously, total-traceability is a sufficient condition for ATPness.

This notion of total-traceability might be a bit hard to detect in general. However, we note it somewhat relates to traceability, in the following sense:

Observation 2.4. *Every traceable graph is total-traceable.*

Proof. This follows from the fact that we can always deduce a total-Hamiltonian path from a Hamiltonian path. Indeed, assume $v_1 \ldots v_n$ is a Hamiltonian path of some graph G. We consider the following sequence $\mathcal{S}$ over the vertices and edges of G. We go through all v_i's in order. Whenever considering a new v_i, we first add v_i to $\mathcal{S}$, and then all forward edges (in arbitrary order) incident to v_i, but with making sure that v_iv_{i+1} (assuming $i < n$) is added last. By *forward edges*, we mean all edges v_iv_j incident to v_i with $i < j$. Then $\mathcal{S}$ yields a Hamiltonian path in $T(G)$, and is thus a total-Hamiltonian path of G. □

We note that a result alike Observation 2.4 cannot hold for edge-traceability. Indeed, a problem is that the order in which edges are traversed in a edge-Hamiltonian path is not always compatible with visiting vertices. This is well exposed when considering stars $K_{1,p}$ with $p \geq 3$ leaves, which are obviously all edge-traceable but not total-traceable.

Along these lines, we observe that graphs being AP, AEP, and ATP at the same time do exist: any path is an obvious example. On the other hand, one can come up with examples of connected[1] graphs that are neither AP, AEP, nor ATP. As an example, just consider any (n, m)-graph G obtained from a star $K_{1,p}$ by subdividing every edge exactly once (resulting in p branches on two vertices and two edges). Assuming p is large enough, it is not too complicated to check that any n-partition or m-partition containing value 3

[1]Obviously, any AP, AEP, or ATP graph must be connected.

at least twice is neither vertex-realisable nor edge-realisable in G. Likewise, any $(n+m)$-partition containing value 5 at least thrice is not total-realisable in G. Thus, assuming p is large enough, G is neither AP, AEP, nor ATP. In next Section 3, we will show, in this vein, that graphs fulfilling some of these properties do not have to fulfil the other ones.

3. Relationships between APness, AEPness, and ATPness

Due to all definitions involved, APness, AEPness, and ATPness are, in spirit to the least, rather close concepts. In this section, we investigate whether having some of these properties necessarily implies having the other ones. In brief, we prove this is not the case, in the sense that for any combination of the three parameters, there are arbitrarily large graphs having these properties, but not having the remaining ones.

Let us recall that Bensmail, in [7], proved that there are arbitrarily large AP graphs that are not AEP, and *vice versa*. Thus, we focus on combinations of the three parameters involving ATPness. That is, we first prove that neither APness nor AEPness implies ATPness, and, *vice versa*, that ATPness implies neither APness nor AEPness. For convenience, we present the four corresponding results in a progressive way (in terms of proof complexity, but also in terms of relationships between the proofs).

Theorem 3.1. *There are arbitrarily large AEP graphs that are not ATP.*

Proof. Just consider any star $G = K_{1,p}$ with $p \geq 4$ leaves. Then G is an (n,m)-graph with $n = p+1$ and $m = p$, and, since $p \geq 4$, we have $n+m \geq 9$. In [7], Bensmail observed that G is edge-traceable, and thus AEP. We claim G cannot be ATP. This follows, in particular, from the fact that G admits no total-realisation of the $(n+m)$-partition $\pi = (3, 3, n+m-6)$. Indeed, note first that $n+m-6 \geq 3$. Thus, all three elements of π have value at least 3, implying that, in any total-realisation of π in G, if we denote by v the center vertex and by $u_1, \ldots, u_p$ the p leaves, having any edge vu_i in some part implies that part must also contain u_i. This implies also that any of the parts of cardinality 3 must contain some u_i, its incident edge u_iv, and v. Since v can belong to only one part but π has at least two elements with value 3, we deduce that a total-realisation of π in G cannot exist. □

Theorem 3.2. *There are arbitrarily large ATP graphs that are not AP.*

Proof. For any $p \geq 3$, we define the *p-comb* as the graph obtained from a path $u_1 \ldots u_p$ on p vertices by adding an edge u_iv_i, where v_i is a new degree-1 vertex, for all $i \in \{2, \ldots, p-1\}$. Let G be any p-comb with $p \geq 3$. Setting $n = |V(G)|$ and $m = |E(G)|$, we have $n = 2p-2 \geq 4$ and $m = 2p-3 \geq 3$, and hence $n+m = 4p-5 \geq 7$. The result follows from the fact that G is never AP, but always ATP. First off, by investigating how matchings behave in G, it is easy to observe that G admits no vertex-realisation of the n-partition $(2, 2, n-4)$; thus, G is never AP. The fact that G is ATP follows *e.g.* from later Lemma 5.2. In brief (see the proof there for more thorough details), we can obtain a total-realisation of any $(n+m)$-partition by essentially adding elements following the sequence $(u_1, u_1u_2, u_2, u_2u_3, u_3, \ldots)$ in order, and, for every $i \in \{2, \ldots, p-1\}$, adding u_iv_i and v_i to some parts after either $u_{i-1}u_i$ or u_i has been added, depending on the number of elements we need to add to the current (partial) part. The claim thus holds. □

In the next result and later on, we get to considering special classes of trees called multipodes, that have been quite investigated in the context of AP trees (see *e.g.* [1, 2, 11, 14]). In brief, *multipodes* are subdivided stars. We will more particularly deal with *tripodes*, which are obtained from claws by subdividing their edges. In particular, for $a, b, c \geq 1$, we

define $T(a,b,c)$ as the tripode obtained from three vertex-disjoint paths $a_1 \dots a_a$, $b_1 \dots b_b$, and $c_1 \dots c_c$ by making all of a_a, b_b, and c_c adjacent to a new vertex r. That is, when removing r from $T(a,b,c)$, what we obtain is three paths of order a, b, and c.

Theorem 3.3. *There are arbitrarily large ATP graphs that are not AEP.*

Proof. Let G be the graph obtained from the tripode $T(2,a,a-2)$ for any $a \geq 8$ and $a \equiv 2 \bmod 3$, by picking any two non-adjacent vertices x and y of the branch with $a-2$ vertices and adding the edge xy, and repeating this process once more. In other words, G is obtained from $T(2,a,a-2)$ by adding two edges along the branch of order $a-2$. Assuming G is an (n,m)-graph, Bensmail observed in [7] that G is not AEP, essentially because G admits no edge-realisation of the m-partition $(3,\dots,3)$ (it can be noted that, indeed, $m \equiv 0 \bmod 3$). We claim that G is ATP. Indeed, a total-realisation $\mathcal{R}$ of any $(n+m)$-partition π can be obtained *e.g.* as follows.

W.r.t. the original tripode, let r be the unique vertex of degree 3, and $a_1,\dots,a_a$, b_1,b_2, and $c_1,\dots,c_{a-2}$ denote the consecutive vertices of the branches of order a, 2, and $a-2$, respectively, where a_1, b_1, and c_1 are the degree-1 vertices (and, thus, a_a, b_2, and c_{a-2} are adjacent to r). We consider the elements of π one by one in any order. For any such element λ, we pick, as a connected part S in G, the first λ elements of the sequence $\mathcal{S} = (a_1, a_1a_2, a_2, a_2a_3, \dots, a_ar)$. That is, we pick elements along the branch of order a, starting from the end-vertex a_1. At any point, if S gets the desired cardinality, then it stands as a connected part of size λ, and we resume with the next element of π as λ. Note that the last element of $\mathcal{S}$ is a_ar; once we have added a_ar to some part S, a few situations can occur:

- If S misses no element, then we mainly pick the remaining connected parts of $\mathcal{R}$ along the ordering $\mathcal{S}' = (b_1, b_1b_2, b_2, b_2r, r, rc_{a-2}, c_{a-2}, c_{a-2}c_{a-3}, c_{a-3}, \dots)$. Recall that, by construction, G has two edges of the form c_ic_j and c_kc_l, where $j < i-1$ and $l < k-1$. In $\mathcal{S}'$, these two edges can be assumed to appear right after c_i and c_k, respectively. Once the process ends, it can be checked that $\mathcal{R}$ is a desired total-realisation.

- The exact same arguments apply if S is missing at least four elements. Indeed, recall that a_ar is adjacent to both rb_2 and rc_{a-2}.

- If S misses exactly one element, then we add r to S, and then pick the remaining parts of $\mathcal{R}$ along $(b_1, b_1b_2, b_2, b_2r, rc_{a-2}, c_{a-2}, c_{a-2}c_{a-3}, c_{a-3}, \dots)$ (where, compared to earlier, r was removed) as above. Note that this has no impact over the connectivity of the parts we pick.

- It remains to consider when S misses exactly two or three elements. Note that, by construction, in $G[\{a_1,\dots,a_a,r\}]$ there are exactly $2a+1$ elements (vertices and edges), while, in $G[\{c_1,\dots,c_{a-2},r\}]$ there are exactly $2(a-2)+1+2 = 2a-1$ elements (remind the two additional edges we have added). Thus, if we build $\mathcal{R}$ as earlier (picking parts following the values in π the exact same order) but along $(c_1, c_1c_2, c_2, c_2c_3, \dots, c_{a-2}r)$ (where, again, the two additional edges are inserted in appropriate places), then, this time, after adding $c_{a-2}r$ to some part S', then:

  - If, after we added a_ar to S, at least three elements were covered by S, then, here, S' contains at least one element, and thus misses four (if S missed two) or five (if S misses three); we then fall into a case similar to one we treated earlier.

– If S contained exactly one or two elements after the addition of $a_a r$, then π contains an element with value in $\{3,4,5\}$. Let us construct $\mathcal{R}$ from scratch. If $3 \in \pi$, then we can pick $\{b_1, b_1b_2, b_2\}$ as a corresponding connected part; if $4 \in \pi$, then we can pick $\{b_1, b_1b_2, b_2, b_2r\}$; while if $5 \in \pi$, then we can pick $\{b_1, b_1b_2, b_2, b_2r, r\}$. In any case, it can be checked that the rest of $\mathcal{R}$ can mainly be picked along $(a_1, a_1a_2, \ldots, a_ar, r, ra_{a-2}, \ldots, a_2a_1, a_1)$ (adding rb_2 after r if this edge remains to be picked, and removing r if this vertex has already been picked). Thus, a total-realisation $\mathcal{R}$ of π in G also exists here.

Thus, $\mathcal{R}$ always exists, and G is ATP. □

Theorem 3.4. *There are arbitrarily large AP graphs that are not ATP.*

Proof. Let us just consider G, a graph obtained from $T(2, a, a-2)$ (for appropriate values of $a \geq 8$) similarly as in the proof of Theorem 3.3 but adding here four edges along the branch of order $a-2$. Now, G is an (n,m)-graph with $n = 2a+1$ and $m = 2a+4$. It was proved that $T(2, a, a-2)$ is AP, see [11], thus G is also AP (since adding edges to an AP graph cannot break its APness). We claim G is not ATP. This is, in particular, because $n+m = 4a+5$ but G admits no total-realisation of the $(n+m)$-partition $\pi = (2a+2, 2a+3)$. Indeed, denoting the vertices of G as in the proof of Theorem 3.3, note that the part S of such a total-realisation $\mathcal{R}$ containing a_1 would have to contain all of $a_1, a_1a_2, a_2, \ldots, a_a, a_ar$, thus at least $2a$ elements. Now, regardless of whether $|S| = 2a+2$ or $|S| = 2a+3$, one of rb_2 or ra_{a-2} must belong to S. Both cases, it cannot be that S covers all of b_1, b_1b_2, b_2, b_2r (recall $a \geq 8$). So, we have that $G - S$ is not connected, and thus $\mathcal{R}$ cannot be a total-realisation. □

We now focus on combinations of properties. Due to the way ATPness is defined, in a way that reminds properties of both APness and AEPness, perhaps the most anticipated question is whether APness and AEPness together imply ATPness. We prove this is not the case, thereby further motivating the study of this latter property. For completeness, we also investigate other combinations, and prove, less surprisingly, that APness and ATPness do not imply AEPness, and similarly that AEPness and ATPness do not imply APness. Again, we present these results in a progressive way.

Theorem 3.5. *There are arbitrarily large graphs that are both AEP and ATP but not AP.*

Proof. Note that p-combs, as introduced in the proof of Theorem 3.2, are edge-traceable, and thus AEP. Since we proved (in Theorem 3.2) that p-combs are ATP but not AP, the result follows. □

Theorem 3.6. *There are arbitrarily large graphs that are both AP and ATP but not AEP.*

Proof. As mentioned earlier, tripodes of the form $T(2, a, a-2)$ with a fulfilling the properties described in the proof of Theorem 3.3 are AP, by a result from [11]. Thus, the graphs G we constructed in the proof of Theorem 3.3 are not only ATP and not AEP (as we proved), but also AP. The result then follows. □

Theorem 3.7. *There are arbitrarily large graphs that are both AP and AEP but not ATP.*

Proof. Consider any of the (n,m)-graphs G from the proof of Theorem 3.4. As we proved, G is AP but not ATP. To be done, it thus remains to prove that G is always AEP. To prove that any m-partition π can be edge-realised in G, we can essentially proceed similarly as in the proof of Theorem 3.3. That is, we add edges to parts following the ordering $(a_1a_2, \ldots, a_ar, rc_{a-2}, \ldots, c_2c_1)$ (where added edges are inserted in appropriate places), and stop as soon as we add a_ar to some part S.

- If S misses at least two edges, then we add b_1b_2 and b_2r to S, before resuming the process along $(rc_{a-2}, c_{a-2}c_{a-3}, \ldots, c_2c_1)$ (where the additional four edges are inserted in appropriate places). Eventually, an edge-realisation results.

- Likewise, if S misses no edge, then we proceed as in the previous case, but adding b_1b_2 and b_2r at the beginning of the ordering of the remaining edges.

- It remains to consider when S misses exactly one edge. Since there are a edges in $G[\{a_1, \ldots, a_a, r\}]$ and $a - 2 + 4 = a + 2$ edges in $G[\{c_1, \ldots, c_{a-2}, r\}]$, if we run the same process but following $(c_1c_2, \ldots, c_{a-2}r, ra_a, \ldots, a_2a_1)$ (where the four additional edges are inserted in appropriate places), then, here, when stopping when $|S|$ edges have been added to parts, we can assume rc_{a-2} and $c_{a-2}c_{a-3}$ are the only edges of $G[\{c_1, \ldots, c_{a-2}, r\}]$ not in parts yet, and the current part, call it S for convenience, misses exactly one edge. We add $c_{a-3}c_{a-2}$ to S so that S gets the desired cardinality. Then the rest of the graph is edge-traceable, and we can thus easily pick the remaining parts in it (just follow any edge-Hamiltonian path).

In all cases, we thus deduce that π can be edge-realised in G, and thus G is also AEP, as claimed. We hence have our conclusion. □

4. Complexity results

In this section, we mostly prove that determining whether an $(n+m)$-partition admits a total-realisation in a given (n, m)-graph is NP-complete, even when restricted to trees. That is, the main decision problem we deal with reads as follows:

Total-Realisation
Instance: An (n, m)-graph G, and an $(n + m)$-partition π.
Question: Is there a total-realisation of π in G?

Before pursuing, let us recall what is known for Vertex-Realisation and Edge-Realisation, the natural vertex and edge counterparts of Total-Realisation.

- Vertex-Realisation is known to remain NP-complete under various restrictions on both G and π, see *e.g.* [5]. In particular, the problem remains NP-complete when π contains 3's only [12] or is of any fixed cardinality at least 2 [5]. Regarding restrictions on G, Vertex-Realisation remains NP-complete *e.g.* when G is a tree with maximum degree at most 3 [2], a subdivided star [9], or a split graph [10].

- Regarding Edge-Realisation, it was proved in [7] that the problem is NP-complete when restricted to trees. By Observation 2.1, this implies that Vertex-Realisation remains NP-complete when restricted to line graphs of trees.

Vertex-Realisation, Edge-Realisation, and Total-Realisation are quite different from the problem of determining whether a given graph is AP, AEP, or ATP. Far less is known about this concern; we postpone a discussion to the concluding section.

As mentioned earlier, we prove below that Total-Realisation is NP-complete, even when restricted to trees. Our proof follows ideas behind analogous results from [7, 9], which is not surprising given Observation 2.1. First off, we need to recall a few facts about the 3-Partition problem, from which our reduction is performed. Recall that this problem, which is well known to be NP-complete (see *e.g.* [13]), is defined as follows:

3-Partition
Instance: A set $A = \{a_1, \ldots, a_{3k}\}$ of size $3k$, a $B \in \mathbb{N}^*$, and an $s : A \to \mathbb{N}^*$ such that:

- $B/4 < s(a) < B/2$ for every $a \in A$, and
- $\sum_{a \in A} s(a) = kB$.

Question: Can A be partitioned into $A_1, \ldots, A_k$ with $\sum_{a \in A_i} s(a) = B$ for all $i \in \{1, \ldots, k\}$?

In particular, we will use the fact that 3-Partition remains NP-complete when restricted to particular types of instances, due to the following result:

Observation 4.1 (Bensmail, Li [9])**.** *Let $<A, B, s>$ be an instance of 3-Partition where:*

- *$B/4 < s(a) < B/2$ for every $a \in A$, and*
- *$\sum_{a \in A} s(a) = kB$.*

The following instances of 3-Partition are equivalent to $<A, B, s>$:

- *$<A, B', s'>$, where $s'(a) = s(a) + 1$ for every $a \in A$, and $B' = B + 3$;*
- *$<A, B'', s''>$, where, for any $\alpha \geq 1$, $s''(a) = \alpha s(a)$ for every $a \in A$, and $B'' = \alpha B$.*

Furthermore, we have:

- *$B'/4 < s'(a) < B'/2$ and $B''/4 < s''(a) < B''/2$ for every $a \in A$, and*
- *$\sum_{a \in A} s'(a) = kB'$ and $\sum_{a \in A} s''(a) = kB''$.*

We are now ready to prove our main result in this section.

Theorem 4.2. *Total-Realisation is NP-complete, even when restricted to trees.*

Proof. Since Total-Realisation is clearly in NP, we focus on proving its NP-hardness. We do it by reduction from the 3-Partition problem, which is NP-complete (see [13]). From an instance $<A, B, s>$ of 3-Partition, we construct, in polynomial time, an (n, m)-graph G (actually a tree) and an $(n + m)$-partition π such that $<A, B, s>$ is a positive instance if and only if G admits a total-realisation of π.

Before describing how to construct G and π, we first need to make sure $<A, B, s>$ fulfils some properties. Namely, we consider each $s(a)$, multiply it by 4, and then add 3 to it. We also modify B, by multiplying it by 4 and adding 9 to it. Clearly, these modifications are achieved in polynomial time. Also, by Observation 4.1, the resulting instance of 3-Partition is equivalent to the original one. Abusing the notation, throughout, for convenience we still deal with the resulting instance as $<A, B, s>$. So, every $s(a)$ fulfils $s(a) \equiv 3 \bmod 4$ and $s(a) \geq 7$, and B is odd.

Let us now describe what G and π are (see Figure 2). We start off with G. We begin from G being a collection of k vertex-disjoint paths $B_1, \ldots, B_k$ on B elements, thus on $\lceil B/2 \rceil$ vertices and $\lfloor B/2 \rfloor$ edges, which is possible since B is odd. For every $i \in \{1, \ldots, k\}$, we choose any end-vertex of B_i, and denote it by u_i. Next, we add a new vertex, r, to the graph, and make it adjacent to all of $u_1, \ldots, u_k$. Finally, for all $i \in \{1, \ldots, k\}$, we add the edge $u_i v_i$, where v_i is a new vertex, and attach at v_i kB new pendant paths of length 2 (thus, not counting v_i, having exactly four elements each: two vertices and two edges). We denote by F_i the $4kB + 2$ elements comprising $v_i u_i$, v_i, and the $4kB$ elements from the kB paths attached at v_i. As for π, we set $\pi = (s_k, s(a_1), \ldots, s(a_{3k}))$, where $s_k = (4kB + 3)k + 1$. Note that π is an $(n + m)$-partition, and that the reduction is done in polynomial time.

The equivalence between $<A, B, s>$ and $<G, \pi>$ follows mainly from a single fact:

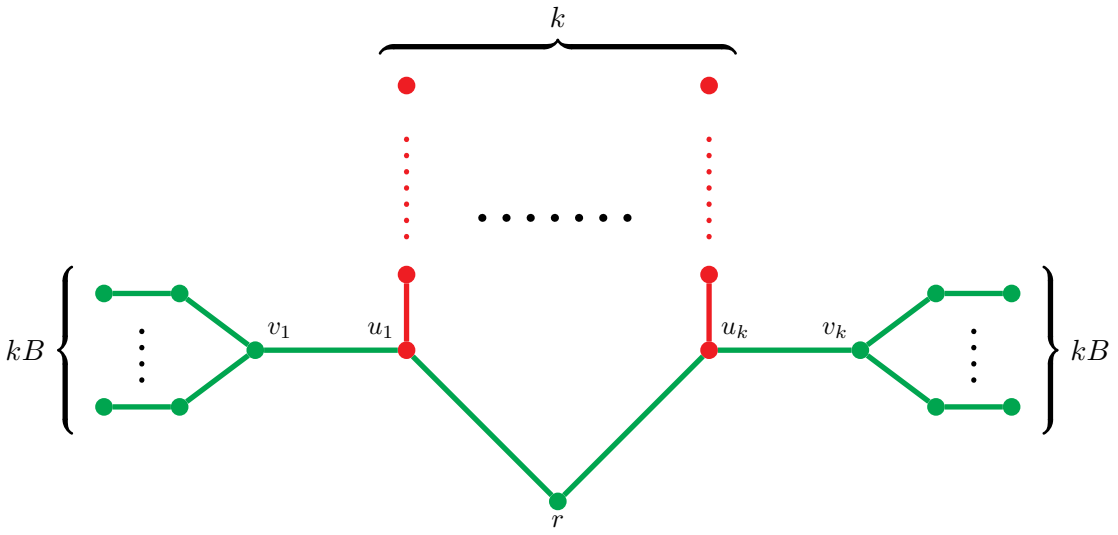


Figure 2: Illustration of the reduced graph in the proof of Theorem 4.2. Elements in green are supposed to be part of the connected part of cardinality s_k in any total-realisation of π in G. Red elements are the other elements, which much be covered by the parts of cardinality $s(a_1), \ldots, s(a_{3k})$.

Claim 4.3. *In any total-realisation $\mathcal{R}$ of π in G, there must be a single part S containing all $4kB+2$ elements of F_i, for all $i \in \{1, \ldots, k\}$.*

Proof of the claim. For convenience, we prove this for $i = 1$. Let us denote by $P_1 = x_1y_1v_1$, $P_2 = x_2y_2v_1, \ldots, P_{kB} = x_{kB}y_{kB}v_1$ the kB paths attached at v_1. Note that if all elements of $P_1, \ldots, P_{kB}$ are covered by a single part S of $\mathcal{R}$, then S has cardinality at least $4kB$, and thus S must be the part of cardinality s_k (as $s(a_1) + \cdots + s(a_{3k}) = kB$). Then, since all elements of π have value at least 7, and $s_k = (4kB+3)k + 1 > 4kB + 2$, it must be that S contains also v_1 and v_1u_1, since v_1u_1 is a bridge of G. Thus, S covers F_1, as claimed.

Now assume that not all elements of $P_1, \ldots, P_{kB}$ are covered by a single part of $\mathcal{R}$. Still because all elements of π have value at least 7, this means there is a part S of $\mathcal{R}$ with $|S| \neq s_k$ such that, say, S contains all of x_1, x_1y_1, y_1, y_1v_1. Note, however, that if, w.r.t., say, P_2, we have only v_2y_2, or only v_2y_2 and y_2, or only v_2y_2, y_2, and y_2x_2 in S, then the remaining elements of P_2 cannot be covered by a connected part of $\mathcal{R}$. This implies that if S contains elements from some P_i, then S must actually cover the four elements of P_i. We know however that $|S| \equiv 3 \bmod 4$. Thus, S must also contain three elements that do not belong to the P_i's. Regardless of whether S contains v_1 or not, note that, by connectivity, S must thus contain v_1u_1. Since F_1 contains $4kB + 2$ elements, $s_k = (4kB + 3)k + 1$, and u_1v_1 is a bridge of G, this implies the part of $\mathcal{R}$ of cardinality s_k cannot cover elements of F_1, which must thus be covered by the other parts only. However, these parts cover $s(a_1) + \cdots + s(a_{3k}) = kB$ elements, while F_1 has $4kB + 2$ elements; a contradiction. ◇

Assume now G admits a total-realisation $\mathcal{R}$ of π. By Claim 4.3, the part S of cardinality s_k must contain the $4kB+2$ elements of F_i for all $i \in \{1, \ldots, k\}$, thereby covering $(4kB+2)k$ elements in total. Now, since ru_i is a bridge of G for all $i \in \{1, \ldots, k\}$, so that S is connected, note that S must also contain all of $ru_1, \ldots, ru_k$. Since r is only incident to these k edges, and all elements of π have value at least 7, we deduce S must contain r as well. Altogether, S must thus cover at least $(4kB+2)k+k+1 = (4kB+3)k+1$ elements; since $|S| = (4kB + 3)k + 1$, we have actually exposed what S exactly is. This means $G - S$, which is nothing but $B_1, \ldots, B_k$, a collection of k vertex-disjoint paths containing B elements each, must be covered by the rest of π, being $(s(a_1), \ldots, s(a_{3k}))$ (that is, $\mathcal{R} - S$ is a total-

realisation of $(s(a_1), \ldots, s(a_{3k}))$ in $G-S$). It should be clear that this is exactly equivalent to solving $<A, B, s>$. Thus, a solution to $<A, B, s>$ can be deduced from $\mathcal{R}$. Conversely, by all the previous arguments, it can be observed that we can deduce a total-realisation of π in G from any solution to $<A, B, s>$. We thus have the claimed equivalence. □

5. Properties of longest paths in ATP graphs

In this section, we investigate how the longest paths of a graph impact on its ATPness. The types of results we establish here are again influenced by previous ones established for APness and AEPness. To name some of the most influential ones:

- Traceable (n, m)-graphs, *i.e.*, having their longest paths going through n vertices, are spanned by the path on n vertices, and are thus AP. However, (n, m)-graphs having their longest paths going through $n-1$ vertices are not necessarily AP (for an example, just note that any tripode $T(1, x, x)$ admits no vertex-realisation of $(x+1, x+1)$). Several authors then strove to push those concerns further, by investigating sufficient conditions, in terms of longest paths and additional properties, guaranteeing a graph is (close to) AP. In particular, Marczyk, in [17], borrowed the parameter σ_2 from Hamiltonicity theory, and studied it in the very context of AP graphs. Recall indeed that, for a graph G,

  $$\sigma_2(G) = \min\{d(u) + d(v) : uv \notin E(G)\},$$

  and that Ore proved, in [22], that any connected (n, m)-graph is Hamiltonian whenever $\sigma_2(G) \geq n$, and traceable whenever $\sigma_2(G) \geq n-1$. Marczyk, in [17], proved that G is AP provided $\sigma_2(G) \geq n-2$ and $\alpha(G) \leq \lceil n/2 \rceil$ (that is, if G has a perfect matching or a quasi-perfect matching). This result was improved upon later on in [15, 18], as Marczyk, Horňák, Schiermeyer, and Woźniak, lowered this sufficient condition to G verifying only $\sigma_2(G) \geq n-5$ and additional conditions (on $\alpha(G)$ and n).

- Regarding AEPness, Bensmail observed in [7] that every connected (n, m)-graph G having its longest path going through at least $n-1$ vertices is always AEP, and that this is not necessarily the case if the longest path goes through $n-2$ vertices only. He also proved that having $\sigma_2(G) \geq n-3$ is a sufficient condition for AEPness.

In what follows, we essentially establish results in this very line, in the context of ATP graphs. More precisely, we prove that the exact sufficient conditions for AEPness also hold for ATPness. For transparency, let us mention that our proofs below are different from those from [7], although, due to the two problems in question being of very close natures, we reuse similar ideas and tools. In particular, one existing tool we need, is the following result of Pósa, which establishes some connection between σ_2 and longest paths.

Theorem 5.1 (Pósa [19])**.** *Let G be a connected graph of order $n \geq 3$ with $\sigma_2(G) \geq \alpha$. If $\alpha < n$, then G contains a path of length α. Otherwise, G is Hamiltonian.*

As a starting point, note that Observation 2.4 implies that every connected (n, m)-graph G with longest path of order n (which occurs in particular if $\sigma_2(G) \geq n-1$) is ATP. We prove first that this also holds true if the longest path is of order at least $n-1$ (and thus when $\sigma_2(G) \geq n-2$). The next result will be useful for that purpose.

Lemma 5.2. *Let G be a connected (n, m)-graph such that $V(G)$ can be partitioned into $I \cup P$, where I is an independent set and $G[P]$ is traceable. If G admits a matching M of cardinality $|I|$ saturating the vertices of I, then G is ATP.*

Proof. Denote by $v_1 \dots v_q$ a Hamiltonian path of $G[P]$, and let $M = \{u_1 v_{i_1}, \dots, u_x v_{i_x}\}$ be a matching of G saturating I, where $I = \{u_1, \dots, u_x\}$ and the v_i's belong to P. To obtain a total-realisation $\mathcal{R}$ in G of any $(n+m)$-partition $\pi = (\lambda_1, \dots, \lambda_p)$, we essentially follow the ideas in the proof of Observation 2.4. That is, we go along $v_1 \dots v_q$, build connected parts on the fly, with the subtlety that, whenever we treat some v_{i_j}, then we also try to incorporate u_j to some part, exploiting the fact that $u_j v_{i_j} \in M$. While this is actually always achievable, there are some preferred ways to do it, which is why we need to describe the process a bit more formally (which will allow for a better understanding later on).

That is, we build the parts of $\mathcal{R}$ one by one, following π. This means we first build the connected part of cardinality λ_1, then that of cardinality λ_2, and so on. For that purpose, we follow the ordering $(v_1, \dots, v_q)$. Whenever considering a new v_i, we pay attention to the number λ of elements of G that must be added to the current (possibly partial) part S. We consider two main cases, depending on whether v_i is incident to an edge of M.

- If v_i is not incident to an edge of M, then we first add v_i to S. In case we had $\lambda = 1$, then note that S has reached the desired cardinality; so, we proceed with building the next part, S', of cardinality λ', if there is indeed one such (otherwise we are done). Let $x \geq 1$ be the number of edges incident to v_i that are not incident to some v_j with $j < i$. Note that these x edges are exactly the edges incident to v_i that have not been added to any part of $\mathcal{R}$ yet. In particular, these x edges might include edges incident to u_i's (but these edges do not belong to M, by assumption).

  - If $x = \lambda'$, then we add these x edges to S', so that S' gets the desired cardinality. The process then goes on with v_{i+1} (if $i < q$) and the next element of π.
  - If $x < \lambda'$, then we add these x edges to S'. To reach the desired cardinality, there remain $\lambda' - x > 1$ elements to be added to S'. The process then goes on with v_{i+1} (which exists), and $\lambda' - x > 1$ elements remaining to be added to S'.
  - If $x > \lambda'$, then we add to S' any λ' of these edges, excluding $v_i v_{i+1}$ (if it exists). We then start picking the next connected parts of $\mathcal{R}$ following π, with the subtlety that $v_i v_{i+1}$, if it exists, must be the last edge incident to v_i added to a part (to ensure its connectivity, upon treating v_{i+1} afterwards).

  In any of these cases, note that the parts we construct are indeed connected by the adding ordering we consider, and that, if v_{i+1} exists, then, when starting treating v_{i+1}, either we start building a new connected part from scratch, or we pursue building a partial part which is connected due to the presence of the edge $v_i v_{i+1}$.

- Assume now v_i is incident to some edge $u_j v_i \in M$. We consider two cases:

  - If $\lambda \geq 2$, then we first add u_j and $u_j v_i$ to S, which preserves that S is connected. Regardless of whether S has reached the desired cardinality, we pursue treating v_i and its remaining incident edges the same way we did in the previous case.
  - If $\lambda = 1$, then we first add v_i to S, so that S is a connected part with the desired cardinality. We then resume the process as in the previous case above, but starting with adding u_j and $u_j v_i$ to parts first.

  Both cases, it can be observed that we again design connected parts, as desired.

Thus, after treating v_q, $\mathcal{R}$ is a total-realisation of π in G. This concludes the proof. □

A direct consequence of Lemma 5.2 is:

Theorem 5.3. *Let G be a connected (n,m)-graph. If the longest path of G goes through at least $n-1$ vertices, then G is ATP. In particular, G is ATP whenever $\sigma_2(G) \geq n-2$.*

Proof. If G is traceable, then the result follows from Observation 2.4. Otherwise, if the longest path $P = v_1 \dots v_{n-1}$ of G goes through exactly $n-1$ vertices, and u is the sole vertex of G not in P, then, because G is connected, any edge uv_i forms a matching (of size 1) saturating $\{u\}$ (which is an independent set). Also, $G[\{v_1, \dots, v_{n-1}\}]$ is traceable. Then G is ATP, by Lemma 5.2. The last part of the claim follows from Theorem 5.1, since having $\sigma_2(G) \geq n-2$ implies the longest path of G goes through at least $n-1$ vertices. □

Theorem 5.3 is best possible, in the sense that there are arbitrarily large connected (n,m)-graphs that are not ATP but have their longest path going through $n-2$ vertices.

Observation 5.4. *There are arbitrarily large non-ATP connected (n,m)-graphs having their longest path going through $n-2$ vertices.*

Proof. For any $x \geq 2$, consider $G = T(x,x,2)$, the tripode where the three branches, B_1, B_2, B_3, not counting the center vertex, r, contain x, x, and 2 vertices, respectively, and thus $2x$, $2x$, and 4 elements, respectively. Thus, G is an (n,m)-graph with $n = 2x+3$, $m = 2x+2$, and $n+m = 4x+5$. Note that, because $x \geq 2$, the longest path of G contains r and all vertices of B_1 and B_2, and thus goes through exactly $n-2$ vertices. To be done with the result, it suffices to observe that G is not ATP, whatever the value of x. Actually, we claim that G admits no total-realisation of the $(n+m)$-partition $(2x+2, 2x+3)$. This is mainly because the part S of such a total-realisation $\mathcal{R}$ containing the end-vertex of, say, B_1 would have to contain, regardless of $|S|$, all $2x$ elements of B_1, and, necessarily, an edge of B_2 or B_3 incident to r, and one or two more elements, depending on whether $|S| = 2x+2$ or $|S| = 2x+3$, respectively. Thus, the rest of G cannot be connected, meaning the second part of $\mathcal{R}$ cannot be connected, a contradiction. Thus, G is not ATP. □

Observation 5.4 justifies, to go farther, to establish sufficient conditions for ATPness in terms of σ_2, rather than solely in terms of the length of the longest path. Namely, to go beyond Theorem 5.3, we prove the following result:

Theorem 5.5. *Let G be a connected (n,m)-graph with $\sigma_2(G) \geq n-3$. Then, either G is isomorphic to $K_{1,4}$ (which is not ATP), or G is ATP.*

Proof. We mostly follow the lines of the proof of Theorem 5.3, taking into account that, by Theorem 5.1, we can mainly assume there are exactly two vertices not in a longest path of G. Namely, by Theorem 5.1, any longest path P of G goes through at least $n-2$ vertices. If P goes through at least $n-1$ vertices, then the result follows from Theorem 5.3. So, we set $P = v_1 \dots v_q$ (where $q = n-2$) and $V(G) \setminus V(P) = \{u_1, u_2\}$. We assume $d(u_1) \geq d(u_2)$.

We consider two main cases, depending on whether u_1u_2 is an edge or not.

- **First case:** $u_1u_2 \notin E(G)$.

  Since G is connected, $d(u_1) = d_P(u_1) \geq 1$ and $d(u_2) = d_P(u_2) \geq 1$ (where, recall, $d_P(u_i)$ denotes the number of neighbours of u_i in P, for every $i \in \{1,2\}$). If $d(u_1) \geq 2$, then we can clearly find, in G, a matching of size 2 saturating $\{u_1, u_2\}$, and the result follows from Lemma 5.2. So, we can now assume $d(u_1) = d(u_2) = 1$. Since u_1 and u_2 are not adjacent, and $\sigma_2(G) \geq n-3$, we have $n \leq 5$. Also, since P is a longest path of G, it cannot be that u_1 or u_2 is adjacent to v_1 or v_q. So, we must have $q = 3$,

thus $n = 5$, and $N(u_1) = N(u_2) = \{v_2\}$. Likewise, we cannot have $v_1v_3 \in E(G)$, as otherwise *e.g.* $v_1v_3v_2u_1$ would be a path of G longer than P, a contradiction. So G must be $K_{1,4}$, which is not ATP since in that case $n+m = 9$ and, obviously, G admits no total-realisation of the 9-partition $(3, 3, 3)$.

- **Second case:** $u_1u_2 \in E(G)$.

  - Assume first u_1 and u_2 have a common neighbour v_i. For some $(n + m)$-partition π, reconsider the core of the building process described in the proof of Lemma 5.2. Let us modify this process by, when reaching v_i, first adding to (partial) parts elements following the sequence $(v_iu_2, u_2, u_2u_1, u_1, u_1v_i)$ before resuming the process from v_i (with v_i not added to a part yet). Eventually, this all results in a total-realisation of π in G. Thus, G is ATP in that case.
  - Assume now u_1 and u_2 have no common neighbours, and, for now, that $d_P(u_2) \geq 1$. Let us denote by v_{i_1} a neighbour of u_1 on P, and by v_{i_2} a neighbour of u_2 on P $(i_1 \leq i_2)$. Then, $v_{i_1} \neq v_{i_2}$. Since P is a longest path of G, we have $i_1, i_2 \notin \{1, q\}$. In particular, both v_{i_1-1} and v_{i_2+1} exist. Also, we can assume $i_2 > i_1 + 2$, as otherwise *e.g.* $v_1 \dots v_{i_1}u_1u_2v_{i_2} \dots v_q$ would be a path of G longer than P.

    If, say, v_{i_1} has a neighbour, v_{i_3}, different from v_{i_1-1} and v_{i_1+1} on P, then we obtain a total-realisation in G of any $(n + m)$-partition π as follows. Free to renaming all vertices of G, we can suppose $i_3 > i_1$. We mostly apply the total-realisation process we described in the proof of Lemma 5.2, with some changes. Namely:

    * when reaching v_{i_1}, if there remains exactly one element to add to the current partial part S, then we add v_{i_1} to the part, and then resume the process but starting with adding to parts elements following the sequence $(u_2, u_2u_1, u_1, u_1v_{i_1})$;
    * if, when reaching v_{i_1}, exactly two elements remain to be added to S, then we add v_{i_1} and $v_{i_1}v_{i_3}$ to S, before resuming the process after, as in the previous case, adding to parts elements following the sequence $(u_2, u_2u_1, u_1, u_1v_{i_1})$;
    * if, when reaching v_{i_1}, exactly three elements remain to be added to S, then we add $v_{i_1}u_1$, u_1, and u_1u_2 to S, before resuming the process, taking into account, since $d_P(u_2) \geq 1$ and $i_1 < i_2$, that u_2 can be added to some part later on, just as in the proof of Lemma 5.2;
    * if, when reaching v_{i_1}, at least four elements remain to be added to S, then we first add all of u_2, u_2u_1, u_1, and $u_1v_{i_1}$ to S, before resuming the process.

    In all cases, it can be checked that we eventually obtain a total-realisation of π in G, as desired. Thus, G is ATP in this case as well.

    The last situation to consider here, is thus when $d_P(v_{i_1}) = 2$, that is, $d(v_{i_1}) = 3$. Likewise, we have $d(v_{i_2}) = 3$. Since, as remarked earlier, we cannot have $i_2 = i_1 + 1$, it cannot be that v_{i_1} and v_{i_2} are adjacent, and, because, $\sigma_2(G) \geq n-3$, we deduce $n \leq 9$. Using computer programs, we were able to observe that, for $n \leq 8$, there are only four connected (n, m)-graphs H in which the longest path P goes through exactly $n-2$ vertices and $\sigma_2(H) \geq n-3$, namely $K_{1,4}$, $K_{2,5}$, $K_{2,5}$ with an additional edge joining the two vertices of degree 5, and the friendship graph F_3 (obtained by having three triangles sharing a vertex). It can be observed that, for any of these four graphs, for any possible P the two vertices not in P share a neighbour, a situation guaranteeing ATPness, as seen earlier. When

$n = 9$, we have $\sigma_2(G) \geq 6$, and thus $d(u_1) \geq 3$, so u_1 has at least two neighbours on P (deduced for instance from the fact that u_1 and v_{i_2} are not adjacent, and $d(v_{i_2}) = 3$). As mentioned earlier, u_1 cannot be adjacent to both v_1 and $v_q = v_7$. Now, since $i_2 > i_1 + 2$, and $i_2 \neq q = 7$, we have $i_2 \in \{5, 6\}$. If $i_2 = 5$, then the neighbours of u_1 cannot lie in $\{v_3, v_4, v_5, v_6\}$, which makes it impossible for u_1 to have two neighbours on P. Thus, $i_2 = 6$, and the previous arguments imply u_1 must have only two neighbours on P, namely v_2 and v_3. But then $v_1 v_2 u_1 v_3 \dots v_7$ is a path of G longer than P, a final contradiction concluding this case.

– Assume last u_1 and u_2 have no common neighbours, and $d_P(u_2) = 0$. Let v_{i_1} be any neighbour of u_1 on P. To begin with, if $d(v_{i_1}) \geq 5$, that is, v_{i_1} is incident to at least two edges e and e' going to P being neither v_{i_1-1} nor v_{i_1+1}, then it can be checked that we can again tweak the process from the proof of Lemma 5.2 to total-realise any $(n + m)$-partition in G. Essentially, it suffices to make sure that e and e' belong to no (partial) parts when reaching v_{i_1} in the process, so that these edges can be used, if needed, to make sure a partial part is completed smoothly. In particular, the most tricky case is, when reaching v_{i_1}, if S misses three elements, in which case we can add all of v_{i_1}, e, and e' to S, before resuming the process but starting with elements following $(u_2, u_2u_1, u_1, u_1v_{i_1})$. If e and e' do not exist, then $d(v_{i_1}) \leq 4$, and, since $d(u_2) = 1$ and $\sigma_2(G) \geq n - 3$ while u_2 and v_{i_1} are not adjacent, we have $n \leq 8$, a case we already discussed.

Thus, in all cases, we deduce G is either $K_{1,4}$, or ATP. □

6. Conclusion

In this work, we have introduced the notion of ATPness, which was intended to have flavours of both APness and AEPness. To investigate how close all these notions are, we reconsidered known, investigated aspects of AP and AEP graphs from literature, and investigated them for ATP graphs. In particular, we established in Section 3 that APness, AEPness, and ATPness are distinct concepts, we provided in Section 4 a complexity result for total-realisations that meets existing ones for vertex-realisations and edge-realisations, and similarly established in Section 5 sufficient conditions for ATPness in terms of longest paths which, for the most part, meet existing sufficient conditions for APness and AEPness.

Although we established that ATPness is different from APness and AEPness in general, a remarkable fact is that ATPness seems closer to APness in some contexts, while it sometimes seems closer to AEPness in others. For instance, regarding our results in Section 4, Vertex-Realisation is known to remain NP-complete for subdivided stars, a result which we were not able to establish regarding Total-Realisation (while the reduced trees in the proof of Theorem 4.2 are closer, in terms of structure, to the reduced trees in the analogous proof for Edge-Realisation in [7]). Meanwhile, Observation 2.4 is an example of result on AP graphs that generalises easily to ATP graphs, but where an analogous result from AEP graphs to ATP graphs does not hold. Still about longest paths, we note that the results we established in Section 5 are closer to existing ones for AEP graphs than for AP graphs. A general feeling we end up with, is that ATPness should be more perceived as AEPness, with vertices both adding constraints but also granting freedom in some cases (just remember how we proved results such as Theorem 5.5, where adding vertices to parts was sometimes performed to solve dead-end situations).

Regarding, notably, our results from Section 3, something valuable we learn is that, although vertex-realisations, edge-realisations, and total-realisations are quite close concepts

(as explained, most of these concepts translate into the other ones, recall *e.g.* Observation 2.1), a crucial point is that, for a given graph, strong discrepancies can show up. This is mainly because of the different notions of connectivity involved, but also because of the fact that having a distinct number of elements to partition, all the while keeping the same graph structure, can open up to new, problematic partitions to consider.

Given our results, we come up with a number of open questions and problems for further work on the topic, which we believe could be worth investigating further.

- Regarding Theorem 4.2, we wonder whether the result also holds when restricting the problem to subdivided stars, a restriction holding for the NP-completeness of the Vertex-Realisation problem [9]. The same question could be asked about the Edge-Realisation problem as well. Likewise, it was proved in [2] that the Vertex-Realisation problem remains NP-complete when restricted to trees with maximum degree 3; it could be interesting to investigate whether a similar result holds for the edge and total counterparts of the problem.

- More generally speaking, an important problem of the field is the complexity of determining whether a given graph is AP. To date, it is unknown whether this problem is complete for some complexity class, and actually it is unclear whether it is even in NP (see [2, 5]), while some partial results are known [2, 5, 9, 10, 20]. We doubt we can go any further for the edge and total versions of the question, but we believe these concerns are of interest and are thus worth recalling.

- Regarding our results in Section 5, we essentially established results matching existing ones for AEPness, thereby showing some discrepancies with analogous results for APness. It could be interesting to investigate whether, regarding both AEPness and ATPness, we can go farther with connected graphs G having lower value of $\sigma_2(G)$. While our proof scheme, relying mainly on Theorem 5.1, still applies, the situation becomes more and more complex as the lower bound decreases. For instance, graphs G with $\sigma_2(G) \geq n-4$ have their longest path on at least $n-3$ vertices, meaning, in the total version, that there might be up to six elements not adjacent/incident to the longest path. While some of our ideas could still be of some use, such as Lemma 5.2, in general we are afraid we could not avoid tedious case distinctions. It might be too that, as the bound decreases, more exceptions appear, just like $K_{1,4}$ for Theorem 5.5.

- An important result when it comes to AP graphs, is that AP trees have bounded maximum degree, namely at most 4 (see [2]). Such a restriction does not hold regarding AEPness, as AEP trees can be of unbounded maximum degree (consider *e.g.* any star, which is edge-traceable and thus AEP). Although we strongly believe ATP trees should be of bounded maximum degree, we were not able to establish such a fact. A piece of evidence we have, is that, by Observation 2.1, partitioning trees in a total way is equivalent to partitioning, in a vertex way, particular graphs having very particular cut-sets. As established notably in [3], it turns out that, for a graph to be AP, its cut-sets must behave in a very gentle way. We suspect this might explain why, perhaps, ATP trees cannot have arbitrarily large maximum degree.

More generally speaking, any interesting question or problem on AP and/or AEP graphs could be worth investigating in the context of ATP graphs.

References

[1] D. Barth, O. Baudon, J. Puech. Decomposable trees: a polynomial algorithm for tripodes. *Discrete Applied Mathematics*, 119(3):205–216, 2002.

[2] D. Barth, H. Fournier. A degree bound on decomposable trees. *Discrete Mathematics*, 306(5):469–477, 2006.

[3] O. Baudon, F. Foucaud, J. Przybyło, M. Woźniak. On the structure of arbitrarily partitionable graphs with given connectivity. *Discrete Applied Mathematics*, 162:381–385, 2014.

[4] O. Baudon, J. Przybyło, M. Woźniak. On minimal arbitrarily partitionable graphs. *Information Processing Letters*, 112:697–700, 2012.

[5] J. Bensmail. On the complexity of partitioning a graph into a few connected subgraphs. *Journal of Combinatorial Optimization*, 30(1):174–187, 2015.

[6] J. Bensmail. Some Properties of Minimal Arbitrarily Partitionable Graphs. *Australasian Journal of Combinatorics*, 86(1):149–168, 2023.

[7] J. Bensmail. Arbitrarily Edge-Partitionable Graphs. Preprint, available online at `https://hal.science/hal-04405426`.

[8] J. Bensmail. A σ_3 condition for arbitrarily partitionable graphs. *Discussiones Mathematicae Graph Theory*, 44(2):755–776, 2024.

[9] J. Bensmail, B. Li. More Aspects of Arbitrarily Partitionable Graphs. *Discussiones Mathematicae Graph Theory*, 42(4):1237–1261, 2022.

[10] H. Broersma, D. Kratsch, G.J. Woeginger. Fully decomposable split graphs. *European Journal of Combinatorics*, 34(3):567–575, 2013.

[11] S. Cichacz, A. Görlich, A. Marczyk, J. Przybyło, M. Woźniak. Arbitrarily vertex decomposable caterpillars with four or five leaves. *Discussiones Mathematicae Graph Theory*, 26:291–305, 2006.

[12] M.E. Dyer, A.M. Frieze. On the complexity of partitioning graphs into connected subgraphs, *Discrete Applied Mathematics*, 10:139–153, 1985.

[13] M.R. Garey, D.S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W.H. Freeman & Co., 1990.

[14] M. Horňák, M. Woźniak. On arbitrarily vertex decomposable trees. *Discrete Mathematics*, 308(7):1268–1281, 2008.

[15] M. Horňák, A. Marczyk, I. Schiermeyer, M. Woźniak. Dense arbitrarily vertex decomposable graphs. *Graphs and Combinatorics*, 28:807–821, 2012.

[16] R. Kalinowski, M. Pilśniak, I. Schiermeyer, M. Woźniak. Dense arbitrarily partitionable graphs. *Discussiones Mathematicae Graph Theory*, 36:5–22, 2016.

[17] A. Marczyk. A note on arbitrarily vertex decomposable graphs. *Opuscula Mathematica*, 26(1):109–118, 2006.

[18] A. Marczyk. An Ore-type condition for arbitrarily vertex decomposable graphs. *Discrete Mathematics*, 309:3588–3594, 2009.

[19] L. Pósa. A theorem concerning Hamiltonian lines. *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, 7:225–226, 1962.

[20] R. Ravaux. *Graphes arbitrairement partitionnables : propriétés structurelles et algorithmiques*. Ph.D. thesis (in French), Université Versailles Saint-Quentin, 2009.

[21] F. Liu, B. Wu, J. Meng. Arbitrarily partitionable $\{2K_2, C_4\}$-free graphs. *Discussiones Mathematicae Graph Theory*, 42:485–500, 2022.

[22] O. Ore. Note on hamilton circuits. *American Mathematical Monthly*, 67:55, 1960.