



Realizable cycle structures in digraphs

Jean Fromentin, P.-L Giscard, T Karaboghossian

► To cite this version:

Jean Fromentin, P.-L Giscard, T Karaboghossian. Realizable cycle structures in digraphs. 2023. hal-03408041v2

HAL Id: hal-03408041

<https://hal.science/hal-03408041v2>

Preprint submitted on 31 May 2023

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

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

Realizable cycle structures in digraphs

Jean Fromentin^a, Pierre-Louis Giscard^{a,*}, Théo Karaboghossian^a

^aUniv. Littoral Côte d'Opale, UR 2597, LMPA, Laboratoire de Mathématiques Pures et Appliquées Joseph Liouville, F-62100 Calais, France

Abstract

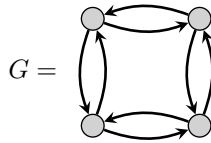
Simple cycles on a digraph form a trace monoid under the rule that two such cycles commute if and only if they are vertex disjoint. This rule describes the spatial configuration of simple cycles on the digraph. Cartier and Foata have showed that all combinatorial properties of closed walks are dictated by this trace monoid. We find that most graph properties can be lost while maintaining the monoidal structure of simple cycles and thus cannot be inferred from it, including vertex-transitivity, regularity, planarity, Hamiltonicity, graph spectra, degree distribution and more. Conversely we find that even allowing for multidigraphs, many configurations of simple cycles are not possible at all. The problem of determining whether a given configuration of simple cycles is realizable is highly non-trivial. We show at least that it is decidable and equivalent to the existence of integer solutions to systems of polynomial equations.

Keywords: walk, simple cycle, trace monoid, hike monoid

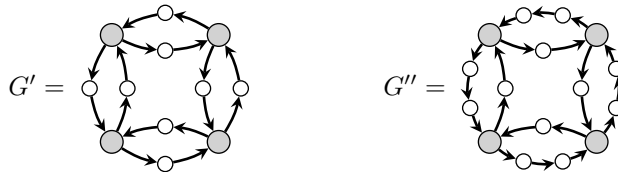
2010 MSC: 05C20, 05C38, 20M05

1. Introduction

The precise nature of the relation between graphs and their walks has, to the best of our knowledge, not yet been thoroughly scrutinized. It seems to be often assumed that walks are slave objects to graphs, in that once the graph is specified its walks can be studied and, in principle, be perfectly known. As a corollary it is expected that properties of the graph leave indelible imprints on its walks, from which the former can thus be inferred. These simple arguments mask the subtle nature of the relation between walks and graphs. For example, consider the following bidirected, vertex-transitive, hence regular, bipartite graph



and let $W_{G:\circ \rightarrow \circ'}$ be the set of all walks from any vertex \circ to any vertex \circ' on G . Now consider the following two digraphs



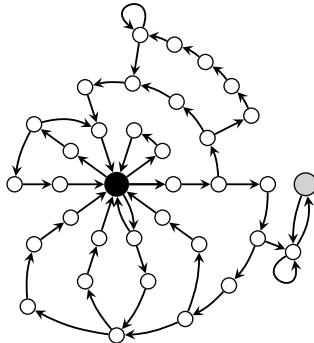
Remark that G' is directed and neither vertex-transitive nor regular. Yet, the set of all walks from any gray vertex to any gray vertex on G' is in bijection with the corresponding set $W_{G:\circ \rightarrow \circ'}$

*Corresponding author

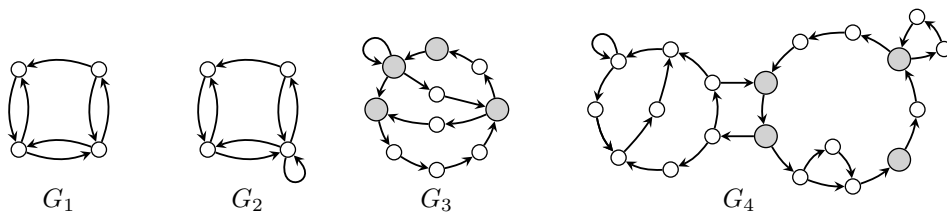
Email address: giscard@univ-littoral.fr (Pierre-Louis Giscard)

on G . The correspondence between both walk sets is simple: the white transient vertices of G' do not add any new cycle to the digraph as compared to G and so only cause the lengths of all of these walks to be multiplied by 2. The structure of every individual walk, i.e., the way it is composed of cycles and of at most one open simple path, is clearly preserved. Rigorously, this makes the bijection into an isomorphism between monoids on walks and walk-like objects, called hikes, something we discuss in details later. Similarly, sets of walks between gray vertices on G'' are in bijection with the corresponding sets on G . In this case however the bijection does not act plainly on the length of individual walks but it continues to be a monoid isomorphism preserving the internal structure of walks. Remarkably here, G'' is not even bipartite. That is, from the point of view of graph theory the transformations from G to G' and from G to G'' are nontrivial and subtle. They correspond to the loss of respectively three and four fundamental graph properties namely bidirectedness, vertex-transitivity, regularity and bipartiteness. Yet walks sets are essentially unchanged by these alterations.

At the very least, these observations suggest a rather loose relationship between graph properties and walk properties. It raises the question of whether there are more graph transformations—excluding graph automorphisms—that preserve walk sets, and if so are they all somewhat trivial, as above with the addition of transient vertices? This question indirectly sprung up in the fields of applied network analysis and machine learning. For example it was noted that vertex centralities, mathematical quantities designed to grasp the relative importance of nodes in a graph, sometimes fail to do so meaningfully [1, 2]. They can for instance predict that an outlying vertex is just as “central” as another one which appears to be at the heart of a graph:



On this digraph with adjacency matrix \mathbf{A} , the gray and black vertices are deemed to be equally central by the subgraph centrality measure $(e^{\mathbf{A}})_{\circ} = (e^{\mathbf{A}})_{\bullet}$, a widely used index of vertex importance in real-world networks [3]. Similarly, measures aimed at grasping the degree of “similarity” between networks are prone to failures. In the field of machine learning these measures, called graph-kernels, are used primarily for tasks of automatic graph classification. In particular it is known that walk-based graph-kernels suffer from the existence of differing graph structures with similar walk counts or arrangement of walks, hindering the automatic distinction of these graph structures, see e.g. [4, 5, 6] and references therein. To illustrate this observation, consider the following four directed graphs



While digraphs G_1 and G_2 might seem most similar with one another among all four, sets of walks between any pairs of gray vertices on graphs G_3 and G_4 are in bijection with sets of walks

on G_1 . As in the earlier example, this bijection preserves the structure of individual walks. At the opposite, there is no such correspondence between the set of all walks on graph G_2 and those on G_1 .

When questioned, these failures have at times been put down to the underlying non-rigorous notions of “importance of a node” and of “similarity between graphs” as we *intuitively* understand them. It is argued that such an understanding, imprecise and fraught with preconceptions, is difficult to express rigorously in mathematical terms thus leading to seemingly absurd results. But the failures could also run deeper and be yet more manifestations of the misappreciated mathematical relation between graphs and walks. Indeed almost all of the measures proposed so far for quantifying such notions as network centrality and graph similarity are algebraic quantities which translate into statements about closed walks. If indeed the cause of the problem is that graphs and their walks give relatively little or at least indirect information about one another, then this issue should be made mathematically rigorous. Considering in particular the case of closed walks on strongly connected digraphs, is perfect knowledge of the former sufficient to determine fundamental properties of the later, possibly up to some triviality?

In this work, we answer the preceding question in the negative: not only are there many transformations between strongly connected digraphs that profoundly alter their properties yet leave their sets of closed walks invariant, but these transformations are diverse and far from trivial. Some produce bijective mappings between walk sets that not only preserve their monoidal structure as in the above examples, but also the length of *all* closed walks. Algebraic quantities routinely used to characterize graphs are left invariant under such mappings. A complete classification of all walk-preserving graph transformations seems to be particularly arduous. The problem finds a wider context in the assertion that a digraph can be reconstructed based solely on structural information about its simple cycles, something which, we find, fails to hold in many a strange way. The complementary question, namely deciding whether a digraph exhibiting certain structural relations between simple cycles exists at all turns out to be unexpectedly difficult. We show at least that this question is decidable and give a meaning for and examples of algebraically closed sets of walks and walk-like objects that cannot, by themselves, be drawn on digraphs. In one more unexpected result we find that these “undrawable” walks do exist on larger digraphs where they are accompanied by a host of algebraically unrelated walks.

In order to talk about closed walks and digraphs in a clearly separate manner so as to untangle their thorny relationship, it is necessary to have a mathematical language for describing sets of closed walks and walk-like objects independently from the digraphs that sustain them. Such a language emerged in a series of contributions starting from the pioneering work by Cartier and Foata on partially commutative monoids [7], through Viennot’s heaps of pieces theory [8, 9] to Giscard and Rochet’s hike monoids [10]. All of the questions presented above then find their natural formulation as statements about trace and hike monoids. This is what we present below. An alternative approach consisting in studying the syntactic structure of walk sets using language theory has also been developed [11, 12, 13, 14].

1.1. The language of closed walks: hike monoids

1.1.1. Notations for graphs and rooted walks

While we begin by recalling standard definitions for graphs, we introduce somewhat less common concepts for walks, of which we advise the reader to take special notice.

A *graph* $G = (V, E)$ is a finite set of vertices V and a finite set E of distinct paired vertices, called edges, denoted $\{i, j\}$, $i, j \in V$. A *digraph* $G = (V, E)$ is a finite set of vertices V and a finite set $E \subseteq V^2$ of *directed edges* (or *arcs*), denoted (i, j) for the arc from i to j . A *directed multigraph* (or *multidigraph*) is defined the same way as a digraph, except that E is a multiset. An edge of E is then denoted $(i, j)_k$, the integer k specifying which edge from i to j we consider. We denote by \mathbf{A} the adjacency matrix of G defined as $\mathbf{A}_{ij} := n$ with $n \geq 0$ the number of directed edges $(i, j)_k \in E$ from vertex $i \in V$ to vertex $j \in V$.

We say that $G' = (V', E')$ is a *subgraph* of a multidigraph G , denoted by $G' \subseteq G$, if $V' \subseteq V$ and $E' \subseteq E$. We say that G' is an *induced subgraph* of a digraph G if furthermore we have $E' = V'^2 \cap E$,

that is G' is obtained from G by deleting some of its vertices and only the edges adjacent to them.

A *rooted walk*, or rooted path, of length ℓ from vertex i to vertex j on a multi directed graph G is a contiguous sequence of ℓ arcs starting from i and ending in j , e.g. $w = (i, i_1)_{k_1} (i_1, i_2)_{k_2} \cdots (i_{\ell-1}, j)_{k_\ell}$ (a sequence of arcs is said to be contiguous if each arc but the first one starts where the previous ended). The rooted walk w is *open* if $i \neq j$ and *closed* otherwise, in which case it is also called *rooted cycle*. A rooted cycle $(i_0, i_1)_{k_1} (i_1, i_2)_{k_2} \cdots (i_{\ell-1}, i_0)_{k_\ell}$ of non-zero length for which all vertices i_t are distinct is said to be *simple*. A self-loop $(i, i)_k$ is considered a rooted simple cycle of length one. On digraphs we may also represent walks unambiguously as ordered sequences of vertices $w = i, i_1, \dots, i_{\ell-1}, j$.

Discarding the piece of information regarding the position of the root turns rooted simple cycles into *simple cycles*. A similar notion concerning rooted closed walks is presented in 1.1.2 An *induced cycle* is a simple cycle for which no pair of visited vertices is linked by an arc that does not belong to the cycle.

A *strongly connected component* of a digraph G is a maximal digraph $G' \subseteq G$ such that for every pair of vertices v_1, v_2 in G' , there is a rooted path in G' from v_1 to v_2 . A digraph G is then said to be *strongly connected* if it is its sole strongly connected component.

1.1.2. Trace monoids and hike monoids

Let Σ be an alphabet then its Kleene star Σ^* designates the set of all finite words on the letters of Σ . Now let $\mathcal{I} \subseteq \Sigma^2$ be a set of pairs of letters. This set defines a rule, called independence relation, which affirms that pairs of letters in \mathcal{I} are independent and can be commuted when they occur next to each other in a word. Thus \mathcal{I} induces an equivalence relation $\sim_{\mathcal{I}}$ between words $w_1, w_2 \in \Sigma^*$ with $w_1 \sim_{\mathcal{I}} w_2$ if and only if it is possible to pass from w_1 to w_2 by commuting adjacent pairs of independent letters. Then the *trace monoid* $\mathcal{T} = \Sigma^* / \sim_{\mathcal{I}}$, is the free partially commutative monoid formed by the $\sim_{\mathcal{I}}$ equivalence classes on Σ^* . These classes are generically called *traces*. At the heart of the trace monoid \mathcal{T} is the partially commutative structure induced by \mathcal{I} . This structure is best represented as a graph, called the *dependency graph* H of \mathcal{T} . It is the graph for which vertices represent all letters of the alphabet Σ and two vertices v_i and v_j are joined by an undirected edge if and only if letters $i, j \in \Sigma$ are *not* allowed to commute per \mathcal{I} . Formally, H is the complement graph of (Σ, \mathcal{I}) .

Trace monoids were introduced by Pierre Cartier and Dominique Foata in their quest for a purely combinatorial proof of MacMahon's master theorem [7, 15]. They considered more specifically the trace monoid on the alphabet of labeled directed edges $\Sigma = E$ of a digraph $G = (V, E)$ with independence relation

$$\mathcal{I}_{\text{CF}} := \{ \{(i, j), (k, l)\} : i \neq k \},$$

where $(i, j) \in E$ and $(k, l) \in E$ are directed edges of G . This rule implies that two directed edges with different starting points are allowed to commute. For example, words $(1, 2)(2, 3)(1, 4)$ and $(2, 3)(1, 2)(1, 4)$ belong to the same Cartier-Foata trace, while $(2, 3)(1, 4)(1, 2)$ belongs to a distinct trace due to the forbidden commutation of $(1, 4)$ with $(1, 2)$ required to pass from the former trace to the latter. This specific instance of trace monoid is sometimes called a *Cartier-Foata monoid*.

Of particular importance for the rest of this study is the submonoid of a Cartier-Foata monoid formed by all traces for which the numbers of incoming and outgoing edges at each vertex are equal. This submonoid is called a *hike monoid* and will be denoted \mathcal{H} . It turns out [10] that hike monoids have a simpler presentation as partially commutative monoids on the alphabet \mathcal{C} of directed graph simple cycles, $\mathcal{H} \simeq \mathcal{C}^* / \mathcal{I}_{\text{Hike}}$, with the independence relation

$$\mathcal{I}_{\text{Hike}} := \{ \{c_1, c_2\} : V(c_1) \cap V(c_2) = \emptyset \},$$

where $V(c_i)$ denotes the set of vertices visited by cycle $c_i \in \mathcal{C}$ and \mathcal{C} is the set of all simple cycles. This rule means that two simple cycles are allowed to commute if and only if they have no vertex in common. We emphasize that simple cycles $c \in \mathcal{C}$ are not rooted, rather they are considered up

to cyclic permutations of their vertices, i.e., their starting point is irrelevant. This is because in the present context a *simple cycle* really is a Cartier-Foata trace and by \mathcal{I}_{CF} two words of E^* where all vertices have one incoming and one outgoing edge and which differ only by their starting point are equivalent. For example, in a digraph $(1, 2)(2, 3)(3, 1) \sim_{\mathcal{I}_{\text{CF}}} (2, 3)(3, 1)(1, 2)$: both words represent the same triangle. By contrast, we verify that the orientation of the simple cycles is preserved by the Cartier-Foata independence relation \mathcal{I}_{CF} since it does not allow reversing edge directions. So for example $(1, 2)(2, 3)(3, 1)$ and $(1, 3)(3, 2)(2, 1)$ are distinct. The dependency graph of a hike monoid will be called the *hike dependency graph*.

Elements of hike monoids are termed *hikes* (Cartier and Foata used the French term “circuits” for the hikes [7], however this name is now widely used in graph theory to designate other objects so [10] adopted the term “hikes” to avoid confusion). Hikes are equivalent classes on words of simple cycles and are mathematically best understood as heaps of such cycles on a graph. In this case, the simple cycles are pieces piled upon one another in such a way that two simple cycles can only be put at the same level if they share no vertex in common. Hikes include *closed walks* as special cases, more precisely it was shown that a hike $h = c_1 \cdots c_k$ whose right-most simple cycle c_k is unique is a walk w in the sense that the words of E^* belonging to the equivalence class w are exactly all the rooted closed walks starting from the vertices of c_k [10]. An arbitrary hike is a walk-like object (hence the name) appearing, when depicted graphically, to be a collection of closed walks.

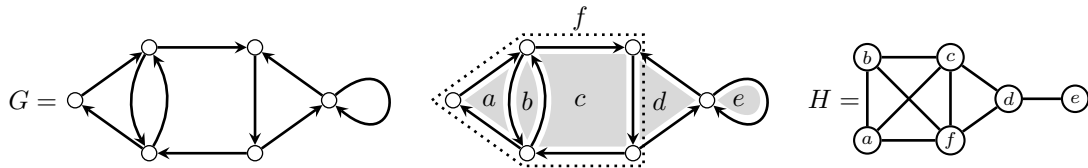


Figure 1: An example of a digraph G (left), the simple cycles on G (middle) and the corresponding hike dependency graph H (right).

2. Preliminaries

With the help of dependency graphs, hike and trace monoids provide a mean of discussing the properties of sets of walks and walk-like objects without explicitly referring to the graph on which these exist. Indeed, the dependency graph H of a hike monoid \mathcal{H} encodes the “cycle skeleton” of a digraph G , as vertices of H are simple cycles on G and edges in H exist whenever two simple cycles of G share at least one vertex. In particular, sets of closed walks and more generally of hikes on two digraphs G and G' sharing the *same* hike dependency graph H are related through a bijection. This bijection stems from the isomorphism between their hike monoids \mathcal{H}_G and $\mathcal{H}_{G'}$. Indeed, these monoids have the same presentation because they have the same number of generators (this being the number of nodes in the hike dependency graph H) as well as the same commutation relations between these generators (the edges in the complement H^c of H). Then \mathcal{H}_G and $\mathcal{H}_{G'}$ are isomorphic and there is a bijection between their sets of words, that is the sets of hikes (and thus of walks) on G and G' . Conversely, the existence of an isomorphism between hike monoids \mathcal{H}_G and $\mathcal{H}_{G'}$ implies that G and G' share the same hike dependency graph. Relating graph properties and walk properties is thus equivalent to understanding which hike dependency graphs do exist and what properties of a digraph can we ascertain from its sole hike dependency graph.

Let ϕ be the map which to any digraph G associates $\phi(G)$, the dependency graph of the hike monoid of G . We say that two digraphs G_1 and G_2 are ϕ -equivalent if $\phi(G_1) = \phi(G_2)$ and we denote this by $G_1 \sim_\phi G_2$. Any digraph transformation passing between two ϕ -equivalent digraphs implements an isomorphism between their hike monoids since these two digraphs share the same hike dependency graph. The problem of understanding the relation between graph properties and walk properties can therefore be formulated as two questions on ϕ :

Question 1 (ϕ -Surjectivity.). *Given a graph H , is there a digraph G such that $\phi(G) = H$?*

If this is the case we say that H is *realizable* and that G *realizes* H . Question 1 is equivalent to asking which trace monoid are hike monoids since any graph H is necessarily the dependency graph of a trace monoid.

Question 2 (ϕ -Injectivity.). *Given a graph H determine $\phi^{-1}(H)$. Equivalently, what are all digraph transformations that induce isomorphisms of hike monoids?*

From the examples presented earlier in the introduction we know already that ϕ is not injective. A good question is thus to characterize the class $\phi^{-1}(H)$ of all digraphs with hike dependency graph H . This, in turn, is equivalent to determining all digraphs transformations that induce isomorphisms of hike monoids since any two digraphs in $\phi^{-1}(H)$ are related by such a transformation.

Observe that hike monoids, map ϕ and questions 1 and 2 naturally extend to multidigraphs. In fact the set of all graphs realizable by multidigraphs is exactly the set of graphs realizable by digraphs since every digraph is a multidigraph and, for any directed multigraph, we can construct a digraph with identical hike monoid by adding transient vertices in the middle of each directed multi-edge. As a consequence, from now on we work equally with multidigraphs and digraphs referring to both as “digraphs”.

For n a positive integer, denote by $[n]$ the set $\{1, \dots, n\}$. Before we address the surjectivity and injectivity questions, we begin with three properties of the map ϕ that reduce considerations to connected dependency graphs H and strongly connected multidigraphs G :

Proposition 2.1. Let G be a digraph. We have

1. If G is strongly connected, then its hike dependency graph $\phi(G)$ is a connected graph.
2. Let G_1, \dots, G_n be the strongly connected components of G . Then $\phi(G) = \bigsqcup_{i \in [n]} \phi(G_i)$.
3. Let H be a connected realizable graph. Then H is realizable by a strongly connected digraph.

Proof.

1. Let c and c' be two simple cycles of G and v and v' respectively be a vertex in c and a vertex in c' . Since G is strongly connected, there exists a cycle containing the vertices v and v' . Hence there exists a sequence of simple cycles c_1, \dots, c_k such that each pairs of simple cycles $\{c, c_1\}, \{c_1, c_2\}, \dots, \{c_{k-1}, c_k\}$ and $\{c_k, c'\}$ share a vertex in common. This implies that the vertices in $\phi(G)$ corresponding to c and c' are in the same connected component, and hence $\phi(G)$ is connected.
2. By definition, if there is a simple cycle in G_1 then this simple cycle is also in G , and if two simple cycles of G_1 share a vertex in common, then they also share a vertex in common in G . Hence we have that $\phi(G_1) \subseteq \phi(G)$. Since this is also true for every $i \in [n]$, we have that $\bigsqcup_{i \in [n]} \phi(G_i) \subseteq \phi(G)$. Let c be a simple cycle of G . Since for every pair of vertices v_1, v_2 in c there is a path from v_1 to v_2 in c , c is a directed subgraph of a connected component G_i of G . Let now be c' an other simple cycle of G which share a vertex in common with c . Then c' is also in G_i , since we can go from any vertex of c to every vertex of c' with a path in $c \sqcup c'$ and vice versa. Hence c and c' are both simple cycles of G_i and they also share a vertex in common in G_i . This implies that $\phi(G) \subseteq \bigsqcup_{i \in [n]} \phi(G_i)$ and so $\phi(G) = \bigsqcup_{i \in [n]} \phi(G_i)$.
3. Let G be a digraph realising H and G_1, \dots, G_n be its strongly connected components. By what precedes, we have that $H = \bigsqcup_{i \in [n]} \phi(G_i)$. Since H is connected, necessarily, there exists $i \in [n]$ such that $H = \phi(G_i)$ and $\phi(G_j) = (\emptyset, \emptyset)$ for every $1 \leq j \neq i \leq n$.

□

We now present some preliminary examples and observations to give the reader a better idea concerning the two questions and the difficulties they present.

First of all, not all graphs are realizable or equivalently, not all trace monoids are hike monoids. The smallest example of a graph that is unrealizable is the square

$$H = \begin{array}{cc} (a) & (b) \\ | & | \\ (d) & (c) \end{array} \quad (1)$$

Let us give an intuitive proof of this statement while we defer a more general formal proof of it to Proposition 2.2. By trying to build a digraph G such that $\phi(G) = H$, we will necessarily construct additional simple cycles. We begin with a simple cycle of arbitrary length corresponding to vertex a (**1.** of Figure 2). We then add the simple cycles corresponding to b and d . They must both share at least one vertex with simple cycle a but not with one another (**2.** of Figure 2). We finish by adding simple cycle c , which must share a vertex with b and d but not with a (**3.** of Figure 2). By doing so we created two new simple cycles, the internal black one and the external gray one (**3.** of Figure 2).

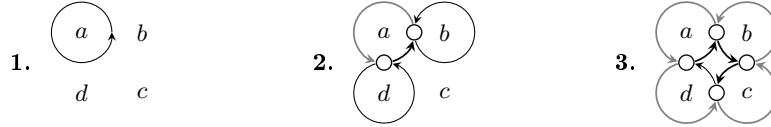


Figure 2: The square graph of (1) is unrealizable: **1.** First consider simple cycle a . **2.** Cycles b and d must intersect a . **3.** Concluding with the construction of d , we obtain 6 simple cycles: a , b , c and d plus the internal black simple cycle and the external gray one.

A natural approach could be to propose that a graph comprising a unrealizable graph as induced subgraph may itself be unrealizable. This cannot be so: consider for example the following graph H comprising the unrealizable square (highlighted with gray edges) as induced subgraph. It turns out that H is realized by the bidirected graph shown on the right,

$$H = \begin{array}{c} \text{A graph with 6 vertices and 12 edges, where a square subgraph is highlighted in gray.} \end{array} \quad \begin{array}{c} \text{A bidirected graph with 4 vertices and 8 edges, representing the realization of H.} \end{array} \in \phi^{-1}(H) \quad (2)$$

We will generalize this observation later, in Remark 3.

This suggests that the problem with realizability lies in part with cycles made of simple cycles, which must themselves be represented as vertices in H . We formalize this intuition and elaborate on it with the following proposition.

Proposition 2.2. Let $H = (V_H, E_H)$ be a realizable graph. Then for any induced simple cycle $C = (c_1, \dots, c_n)$ of length at least 4 in H , there exist two vertices $w_1, w_2 \in V_H$ such that:

- for all vertices c_i in C , edges $\{w_1, c_i\}$ and $\{w_2, c_i\}$ are both in E_H .
- the neighbourhoods of w_1 and w_2 are included in the neighbourhood of C . That is to say if $\{v, w_1\} \in E_H$ or $\{v, w_2\} \in E_H$ then either v is in C or there exists c_i in C such that $\{v, c_i\} \in E_H$.

Furthermore, if edge $\{w_1, w_2\}$ is not in H then for any $i \in [n]$ there exists a vertex c'_i of H such that edges $\{c_i, c'_i\}$ and $\{c_{i+1}, c'_i\}$ are in E_H , under the convention $c_{n+1} = c_1$.

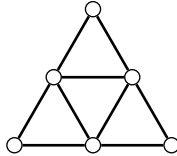
Proof. Let $C = (c_1, \dots, c_n)$ be an induced simple cycle of H of length $n \geq 4$ and let $G = (V_G, E_G)$ be a digraph realizing H . Let $v_1, v_2 \in V_G$ be two vertices visited by c_1 such that v_1 is also visited by c_n , v_2 is also visited by c_2 and every other vertex in the path from v_1 to v_2 along c_1 is neither visited by c_n nor by c_2 . For $i = 3$ up to n , we recursively define v_i as the first vertex in c_i we encounter when walking along cycle c_{i-1} from v_{i-1} . For $i \in [n]$ and $v, v' \in c_i$, let us denote by $v \rightarrow_i v'$ the path from v to v' along c_i (if $v = v'$ then the path has no edge) and let us show that the rooted closed walk $w_1 = v_1 \rightarrow_1 v_2 \rightarrow_2 v_3 \dots v_n \rightarrow_n v_1$ is a rooted simple cycle.

First, w_1 comprises at least one edge since $n \geq 4$ and C being an induced cycle entails $v_1 \neq v_3$. Second, suppose that w_1 is not simple and let v be a vertex visited at least twice by w . Let $i \in [n]$ (under the convention $v_0 = v_n$ and $v_1 = v_{n+1}$) be such that v occurs in $v_i \rightarrow_i v_{i+1}$, meaning in particular that v is visited by c_i . Since C is an induced cycle of H , any other appearance of v must be in $v_j \rightarrow_j v_{j+1}$ with $j = i - 1$ or $j = i + 1$, as otherwise $\{c_j, c_i\}$ would be a chord of C . Assuming $j = i - 1$ without loss of generality, v is a vertex visited by c_i which also occurs in $v_{i-1} \rightarrow_{i-1} v_i$. But if $i \neq 1$, by construction, v_i is the only vertex of $v_{i-1} \rightarrow_{i-1} v_i$ which is also visited by c_i , implying $v = v_i$ is the only occurrence of v , contradicting our supposition that w is not simple. If $i = 1$, then again by construction v_1 is the only vertex of $v_1 \rightarrow_1 v_2$ which is also visited by c_n , hence $v = v_1$ and we have the same conclusion. Consequently, w_1 describes a simple cycle satisfying the two claims of the Proposition.

Let $v'_1, v'_n \in V_G$ be two vertices visited by c_n such that v'_1 is also visited by c_1 , v'_n is also visited by c_{n-1} and every other vertex in the path from v'_1 to v'_n along c_n is neither visited by c_1 nor by c_{n-1} . For $i = n - 1$ down to 2, we recursively define v'_i as the first vertex visited by c_{i-1} we encounter when walking along cycle c_i from v'_{i+1} . By the same reasoning as for w_1 , walk $w_2 = v'_1 \rightarrow_n v'_n \rightarrow_{n-1} v'_{n-1} \dots v'_2 \rightarrow_1 v'_1$ is a simple cycle satisfying the two claims of the Proposition. Furthermore, w_2 crosses cycles c_1, \dots, c_n in reverse order as compared to w_1 and so is distinct from it.

We now turn to the situation where edge $\{w_1, w_2\}$ is not in E_H , i.e., w_1 and w_2 have no vertex in common in G . Let $i \in [n]$. By construction, vertices v_i and v'_i are both in c_i and c_{i-1} and are distinct since v_i is in w_1 and v'_i is in w_2 . Let us consider cycle $\mathbf{c} := v_i \rightarrow_{i-1} v'_i \rightarrow_i v_i$ and show that v_{i-1} can not be visited by it. First, v_{i-1} can not be visited by path $v'_i \rightarrow_i v_i$ since it is in c_{i-2} , which is disjoint from c_i . It can also not be in path $v_i \rightarrow_{i-1} v'_i$ as otherwise v'_i would be in $v_{i-1} \rightarrow_{i-1} v_i$, which is a part of w_1 . Then v_{i-1} is not visited by \mathbf{c} . By the same reasoning we find that v'_{i+1} is not visited by \mathbf{c} either. As a consequence \mathbf{c} is distinct from c_{i-1} and c_i and shares at least one vertex with each of them. If \mathbf{c} is simple then we choose $c'_i := \mathbf{c}$ equal to this cycle, else we choose c'_i to be a simple cycle composing \mathbf{c} and visiting v_i . \square

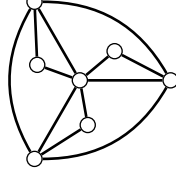
Triangles themselves are no obstacle to realizability as any triangle is realizable by a bouquet of three self-loops on the same vertex. This does not entail that any chordal graph is realizable however. For example, the trifold graph:



is unrealizable for the same reasons barring the square from being realized. In fact, this graph can be seen as the particular case of Proposition 2.2 for cycles of length 3. Indeed, the reader may attempt to build a digraph with a trifold dependency graph to see that because of the 3 external vertices, the cycles composing the central triangle necessarily imply the existence of two further cycles sharing vertices with every other cycle hence a pair of vertices connected to all vertices of the trifold, contradicting its structure.

All observations made so far point to realizability as depending solely on the induced cycles of H and their neighborhood, with the need to distinguish long (length $\ell \geq 4$) from short ones (exception of the trifold graph). This is in fact not true; for example we prove later in Appendix A

that the following graph is unrealizable yet possesses neither an induced cycle of length $\ell \geq 4$ nor a triform of triangles:



3. Surjectivity: which graphs are realizable?

As shown by the preliminary examples and results of Section 2, understanding the image of ϕ is more complicated than it may first appear. Ideally, one would like to have a criterion for deciding if a graph is realizable, i.e., if it is in the image of ϕ . Such a criterion remains elusive and if it exists, it must be highly non-trivial as we will demonstrate. Nonetheless we here establish that realizability is *decidable* by providing an algebraic condition that is equivalent to it. While this algebraization does not in itself shed additional light on realizability it leads to an algorithm for systematically checking for it.

Given a digraph $G = (V, E)$ and $H = \phi(G)$ its hike dependency graph we have, by definition, that the vertices of H correspond to the simple cycles of G . Conversely we may ask under what form do the vertices of G manifest themselves in H ? Given v a vertex of G , all simple cycles of G visiting v share at least this vertex hence do not commute in \mathcal{H}_G . This implies that they are all mutually connected by edges in H , i.e., they form a clique κ_v . The fact that vertices of G yield cliques in H leads to the following observations:

- i) The set $\{\kappa_v\}_{v \in V}$ is a *clique cover* of H , that is each clique κ_v is a subgraph of H and every edge and vertex of H appears in at least one κ_v .
- ii) For $W \subseteq V$, the set $\bigcap_{v \in W} \kappa_v \setminus \bigcup_{v \in V \setminus W} \kappa_v$ corresponds to the simple cycles of G with exactly W as vertex set.

Let us now give a criterion equivalent to realizability. For S a set, we denote by \mathfrak{C}_S the set of permutations over S with cycle decomposition of length 1.

Theorem 3.1. Let H be a graph. Then H is realizable if and only if there exists a clique cover $\{\kappa_1, \dots, \kappa_n\}$ of H such that the following polynomial system in variables $(m_{ij})_{i,j \in [n]}$ admits an integer solution:

$$\forall W \subseteq [n], \quad \sum_{\sigma \in \mathfrak{C}_W} \prod_{v \in W} m_{v, \sigma(v)} = |\mathcal{K}_W|, \quad (3)$$

where $\mathcal{K}_W := \bigcap_{v \in W} \kappa_v \setminus \bigcup_{v \in [n] \setminus W} \kappa_v$.

In this case, H is realized by the digraph G with vertex set $[n]$ and adjacency matrix \mathbf{A} defined by $\mathbf{A}_{i,j} := m_{i,j}$ for $1 \leq i, j \leq n$.

Corollary 3.2. Realizability is decidable.

Proof of Theorem 3.1. Let us first suppose that H is realizable and let G be a digraph realizing H with vertex set $[n]$ and adjacency matrix \mathbf{A} . From observation i) above, set $\{\kappa_v\}_{v \in [n]}$ is a clique cover of H . Furthermore, from ii) it follows that for any given subset $W \subseteq [n]$, the number of cycles with vertex set W is equal to $|\mathcal{K}_W|$. Now remark that this number is also equal to $\sum_{\sigma \in \mathfrak{C}_W} \prod_{v \in W} \mathbf{A}_{v, \sigma(v)}$. Indeed, to choose a simple cycle with vertex set W in G , we must first choose in which cyclic order we visit the vertices in W , that is a permutation in \mathfrak{C}_W , and then, for every vertex $v \in W$ choose which edge to follow from v to $\sigma(v)$ among the $\mathbf{A}_{v, \sigma(v)}$ possibilities. Hence system (3) admits $m_{i,j} = \mathbf{A}_{i,j}$ as an integer solution.

Let us now suppose that system (3) admits an integer solution and let G be the digraph with vertex set $[n]$ and adjacency matrix \mathbf{A} such that $\mathbf{A}_{i,j} = m_{i,j}$ for $1 \leq i, j \leq n$. Let $H' := \phi(G)$ be

the hike dependency graph of G and for every $v \in [n]$ let κ'_v be the clique in H' corresponding to the simple cycles of G visiting v . As for \mathcal{K}_W , we define \mathcal{K}'_W to be $\bigcap_{v \in W} \kappa_v \setminus \bigcup_{v \in [n] \setminus W} \kappa_v$. Then from what precedes, we know that for any $W \subseteq [n]$, the number of vertices in \mathcal{K}'_W is equal to $\sum_{\sigma \in \mathcal{C}_W} \prod_{v \in W} \mathbf{A}_{v, \sigma(v)}$. But since $(\mathbf{A}_{i,j})_{i,j \in [n]} = (m_{i,j})_{i,j \in [n]}$ is a solution of (3), this is also equal to the number of vertices in \mathcal{K}_W . Consequently, there is a bijection ψ_W between these two sets \mathcal{K}_W and \mathcal{K}'_W for every $W \subseteq [n]$. In addition, because $(\mathcal{K}'_W)_{W \subseteq [n]}$ and $(\mathcal{K}_W)_{W \subseteq [n]}$ are partitions of the vertex sets of H' and of H , respectively, there exists a global bijection ψ between the vertices of H' and of H . In the same way, since for any $u \in [n]$ the sequences $(\mathcal{K}'_W)_{u \in W \subseteq [n]}$ and $(\mathcal{K}_W)_{u \in W \subseteq [n]}$ are partitions of κ'_u and of κ_u , respectively, ψ realizes a bijection between κ'_u and κ_u . This makes ψ into an isomorphism of graphs. Indeed, let be v_1 and v_2 be two vertices of H , then we have

$$\begin{aligned} \text{edge } \{v_1, v_2\} \text{ is in } H' &\iff \exists v \in [n], v_1, v_2 \in \kappa'_v \\ &\iff \exists v \in [n], \psi(v_1), \psi(v_2) \in \kappa_v \\ &\iff \text{edge } \{\psi(v_1), \psi(v_2)\} \text{ is in } H. \end{aligned}$$

Whence H is realizable and G realizes H . \square

Before giving examples of Theorem 3.1, let us mention that we can ignore clique covers such that $\kappa_i \subseteq \kappa_j$ for some $i \neq j$. Indeed, if for such a clique cover the system (3) admits an integer solution, then no cycles would pass by the vertex i in the associated digraph, since we have $|\mathcal{K}_W| = 0$ for any W containing i . Hence the digraph obtained by removing vertex i still realizes H and corresponds to the clique cover obtained by removing the clique κ_i . In the sequel, we call trivial any clique κ_i such that $\kappa_i \subseteq \kappa_j$ for some $j \neq i$.

Let us now illustrate the theorem with three concrete cases.

Examples 3.3.

- Let H be the square over $\{a, b, c, d\}$ as in (1).

The only clique cover of H with no trivial clique is $\kappa_1 = \{a, b\}$, $\kappa_2 = \{b, c\}$, $\kappa_3 = \{c, d\}$ and $\kappa_4 = \{d, a\}$. System (3) comprises an equation for each subset W of $\{1, 2, 3, 4\}$. Firstly, considering $W = \emptyset$ we get the trivial equation $0 = 0$. Secondly, for $W = \{1\}$, we find

$$m_{1,1} = |\kappa_1 \setminus (\kappa_2 \cup \kappa_3 \cup \kappa_4)| = |\{a, b\} \setminus \{a, b, c, d\}| = 0,$$

and repeating the same reasoning for $W = \{2\}, \{3\}$ and $\{4\}$ we obtain

$$m_{1,1} = m_{2,2} = m_{3,3} = m_{4,4} = 0.$$

Thirdly, for $W = \{1, 2\}$, we find $m_{1,2}m_{2,1} = |(\kappa_1 \cap \kappa_2) \setminus (\kappa_3 \cup \kappa_4)| = |\{b\} \setminus \{d\}| = 1$. Similarly $W = \{2, 3\}, \{3, 4\}$ and $\{1, 4\}$ yield

$$m_{1,2}m_{2,1} = m_{2,3}m_{3,2} = m_{3,4}m_{4,3} = m_{1,4}m_{4,1} = 1. \quad (4)$$

As $\kappa_1 \cap \kappa_3 = \kappa_2 \cap \kappa_4 = \emptyset$, the set \mathcal{K}_W is empty whenever $\{1, 3\}$ or $\{2, 4\}$ is a subset of W :

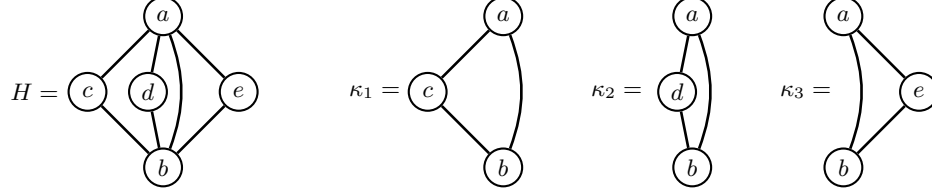
$$\begin{aligned} m_{1,3}m_{3,1} &= m_{2,4}m_{4,2} = 0 && \text{for } W = \{1, 3\}, \{2, 4\}, \\ m_{1,2}m_{2,3}m_{3,1} + m_{1,3}m_{3,2}m_{2,1} &= 0 && \text{for } W = \{1, 2, 3\}, \\ m_{1,2}m_{2,4}m_{4,1} + m_{1,4}m_{4,2}m_{2,1} &= 0 && \text{for } W = \{1, 2, 4\}, \\ m_{1,3}m_{3,4}m_{4,1} + m_{1,4}m_{4,3}m_{3,1} &= 0 && \text{for } W = \{1, 3, 4\}, \\ m_{2,3}m_{3,4}m_{4,2} + m_{2,4}m_{4,3}m_{3,2} &= 0 && \text{for } W = \{2, 3, 4\}. \end{aligned}$$

Lastly, for $W = \{1, 2, 3, 4\}$, we get

$$m_{1,2}m_{2,3}m_{3,4}m_{4,1} + m_{1,2}m_{2,4}m_{4,3}m_{3,1} + m_{1,3}m_{3,2}m_{2,4}m_{4,1} + m_{1,3}m_{3,4}m_{4,2}m_{2,1} = 0. \quad (5)$$

This system does not have any integer solution since (4) forces $m_{1,2} = m_{2,3} = m_{3,4} = m_{4,1} = 1$. In particular $m_{1,2}m_{2,3}m_{3,4}m_{4,1}$ equals 1, which is impossible by (5). We recover the observation that the square is unrealizable.

- Let H be the following graph over $\{a, b, c, d, e\}$ with clique cover $\{\kappa_1, \kappa_2, \kappa_3\}$:



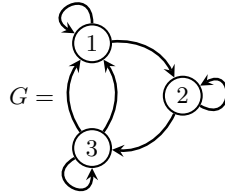
Then, considering $\emptyset \neq W \subseteq \{1, 2, 3\}$ we have

W	\mathcal{K}_W
$\{1\}$	$\kappa_1 \setminus (\kappa_2 \cup \kappa_3) = \{c\}$
$\{2\}$	$\kappa_2 \setminus (\kappa_1 \cup \kappa_3) = \{d\}$
$\{3\}$	$\kappa_3 \setminus (\kappa_1 \cup \kappa_2) = \{e\}$
$\{1, 2\}$	$(\kappa_1 \cap \kappa_2) \setminus \kappa_3 = \emptyset$
$\{1, 3\}$	$(\kappa_1 \cap \kappa_3) \setminus \kappa_2 = \emptyset$
$\{2, 3\}$	$(\kappa_2 \cap \kappa_3) \setminus \kappa_1 = \emptyset$
$\{1, 2, 3\}$	$\kappa_1 \cap \kappa_2 \cap \kappa_3 = \{a, b\}$

and system (3) is given by:

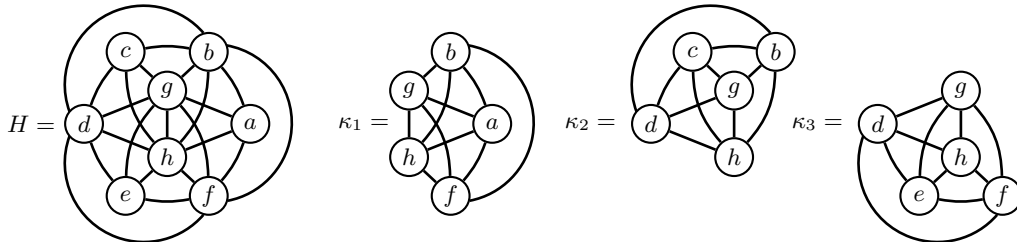
- i) $m_{1,1} = m_{2,2} = m_{3,3} = 1$,
- ii) $m_{1,2}m_{2,1} = m_{2,3}m_{3,2} = m_{3,1}m_{1,3} = 0$,
- iii) $m_{1,2}m_{2,3}m_{3,1} + m_{1,3}m_{3,2}m_{2,1} = 2$.

This system admits the following solution: $m_{1,1} = m_{2,2} = m_{3,3} = m_{1,2} = m_{2,3} = 1$, $m_{3,1} = 2$ and $m_{1,3} = m_{3,2} = m_{2,1} = 0$, as well as 5 further solutions (obtained on changing which of the monomials $m_{1,2}m_{2,3}m_{3,1}$ and $m_{1,3}m_{3,2}m_{2,1}$ is null and which variable $m_{i,j}$ is equal to 2). This indicates that H is realizable by digraph G with adjacency matrix $\mathbf{A}_{i,j} = m_{i,j}$, that is



for which we verify that $\phi(G) = H$ holds as predicted.

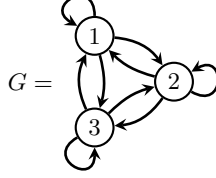
- Let H be the following graph over $\{a, b, c, d, e, f, g\}$ with clique cover $\{\kappa_1, \kappa_2, \kappa_3\}$:



The system (3) is given by:

- i) $m_{1,1} = m_{2,2} = m_{3,3} = 1$,
- ii) $m_{1,2}m_{2,1} = m_{2,3}m_{3,2} = m_{3,1}m_{1,3} = 1$,
- iii) $m_{1,2}m_{2,3}m_{3,1} + m_{1,3}m_{3,2}m_{2,1} = 2$.

This system admits the solution: $m_{i,j} = 1$ for every $1 \leq i, j \leq 3$ The digraph G with adjacency matrix $\mathbf{A}_{i,j} = 1$ is then:



and we find that H is the hike dependency graph of G as requested.

We can deduce the following corollary from Theorem 3.1.

Corollary 3.4. All trees are realizable.

Proof. Let $T = (V, E)$ be a tree. The cliques of a tree are exactly its edges and its vertices, hence the unique clique cover of T with no trivial clique is its set of edges $E = \{e_1, \dots, e_n\}$. For each $v \in V$, denote by $E(v)$ the set of edges containing v . The polynomial system (3) is then given by:

$$\forall W \subseteq E, \quad \sum_{\sigma \in \mathfrak{C}_W} \prod_{e \in W} m_{e, \sigma(e)} = \begin{cases} 1 & \text{if } \exists v \in V, W = E(v) \\ 0 & \text{else} \end{cases}.$$

A solution of this system consist in choosing a permutation $\sigma_v \in \mathfrak{C}_{E(v)}$ for each $v \in V$ and assigning 1 to each $m_{e, \sigma_v(e)}$ for $e \in E(v)$ and 0 to all others $m_{i,j}$. \square

Remark 1. The digraph obtained in the above proof realizing a tree T is a directed medial graph of T .

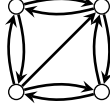
As mentionned in Corollary 3.2, realizability is decidable: an algorithm checking for it can work through all possible cliques covers of a graph H , verifying for each such cover if the accompanying system (3) admits an integer solution. In practice, the search for cliques covers can be highly time consuming since we are searching all the possible ways to cover a set (the edges of H) with a set of subsets (the cliques). This algorithm could be greatly accelerated by restricting the pool of cliques covers. In fact, in Example 3.3, we always presented a clique cover with minimal number of cliques. This seems to be always possible hence the following conjecture.

Conjecture 3.5. Let H be a realizable graph. Then we can always find a digraph G realizing H with number of vertices $|V_G|$ equal to the smallest amount of cliques necessary to cover H . That is to say, Theorem 3.1 continues to hold if we require that n be minimal.

The fact that finding a clique cover with minimal number of cliques such that system (3) has an integer solution implies realizability is true. It is the converse assertion that is difficult to prove: does realizability always imply the existence of a *minimal* clique cover? The natural approach to prove this would be to suppose that a graph H is realizable by a digraph G and then transform G so as to preserve its hike dependency graph while lowering its number of vertices. This is non-trivial and directly related to the question of ϕ -injectivity which we study in the next section.

Remark 2. Contrary to the second and third cases of Examples 3.3, even with Conjecture 3.5 true, the clique cover leading to an integer solution of (3) is not necessarily made of maximal cliques. For $H_1 = (V_1, E_1)$ and $H_2 = (V_2, E_2)$ two graphs, let us denote by $H_1 \bowtie H_2$ the graph with vertex set $V_1 \sqcup V_2$ and edge set $E_1 \sqcup E_2 \sqcup (V_1 \times V_2)$. Let H_1 be the square graph on vertices

$\{a, b, c, d\}$, H_2 be the complete graph on 4 vertices $\{e, f, g, h\}$ and H be the graph $H_1 \bowtie H_2$. On H , the minimal size of a clique cover is 4, since every clique is a subgraph of a maximal clique and H has 4 maximal cliques: $\{a, b, e, f, g, h\}$, $\{b, c, e, f, g, h\}$, $\{c, d, e, f, g, h\}$ and $\{d, a, e, f, g, h\}$. One can check that system (3) does not have an integer solution if we choose the maximal cliques to form the clique cover, yet it has an integer solution if we choose instead cliques $\{a, b, e, f, g\}$, $\{b, c, e, f, g, h\}$, $\{c, d, e, f, h\}$ and $\{d, a, e, f, g, h\}$ to form the clique cover. This solution leads to the following digraph:



Using Proposition 2.2 to test for non-realizability, Conjecture 3.5 to test for realizability and with ad-hoc arguments in 9 remaining undecided cases, we computed the number of unlabelled connected realizable graphs with up to 7 vertices as (OEIS A348365, computations based on the list of unlabelled connected graphs available at <http://users.cecs.anu.edu.au/~bdm/data/graphs.html> and [16])

$$1, 1, 2, 5, 15, 58, 265.$$

In comparison, the number of unlabelled connected graphs on up to 7 vertices is known to be (OEIS A001349)

$$1, 1, 2, 6, 21, 112, 853.$$

Whilst most graphs are realizable for small number of vertices this changes when more vertices are considered. We conjecture that the proportion of realizable graphs on n vertices with respect to all unlabelled connected graphs on n vertices decreases as n increases. At the same time, we know that:

Proposition 3.6. Let h_n be the number of connected realizable graphs on $n \geq 1$ vertices. Then the sequence $\{h_n\}_{n \geq 1}$ is strictly increasing, and grows at least exponentially with n as $n \rightarrow \infty$.

Proof. Let H_1 and H_2 be two realizable graphs on $n \geq 1$ and $m \geq 1$ vertices, respectively, and let $G_1 \in \phi^{-1}(H_1)$ and $G_2 \in \phi^{-1}(H_2)$. Let v_1 be a vertex of G_1 and v_2 be a vertex of G_2 and consider the glued digraph G_3 obtained by taking the disjoint union of G_1 and G_2 and fusing v_1 with v_2 . Then $H_3 = \phi(G_3)$ is obtained by taking the disjoint union of H_1 and H_2 and adding an edge between each vertex of H_1 corresponding to a cycle containing v_1 and each vertex of H_2 corresponding to a cycle containing v_2 . In particular, since no simple cycle was created in G_3 that is not already in G_1 or G_2 , H_3 has $n + m$ vertices and $h_{n+m} \geq h_n h_m$. From this we take the logarithm to get $\log h_{n+m} \geq \log h_n + \log h_m$, i.e., the sequence $\{\log h_n\}_{n \geq 1}$ is superadditive. Then by Fekete's lemma $\log \eta := \lim_{n \rightarrow \infty} n^{-1} \log h_n$ exists and is given by $\sup_{n \geq 1} n^{-1} \log h_n$. If this sup is finite, η is a finite constant and h_n grows exponentially as $n \rightarrow \infty$. If the sup is infinite then h_n grows superexponentially as $n \rightarrow \infty$.

Second, setting $m = 1$ leads to $h_{n+1} \geq h_n h_1 = h_n$ as there is only one graph H on one vertex, which is realized by the graph G_1 made of a single simple cycle c_1 of arbitrary length. Now consider the cone graph H_{n+1} comprising an induced cycle on $n - 1$ vertices with two additional vertices connected to all others. This graph cannot be obtained as the hike dependency graph of a digraph obtained by gluing G_1 to another digraph G' . Indeed, c_1 can neither be one of the two summits of the cone because by Proposition 2.2, $H_{n+1} \setminus c_1$ would not be realizable; nor can c_1 be a part of the long induced cycle in H_{n+1} as all neighbors of c_1 would form a clique. Then $h_{n+1} \geq h_n + 1 > h_n$ and the sequence is strictly increasing. \square

The non-realizability of most of the graphs on up to 7 vertices is decided thanks to Proposition 2.2, which relies on induced cycles of simple cycles. Overall, when evaluating h_7 , we are left with 9 graphs for which ad-hoc arguments must be found as their realizability could neither be invalidated by the Proposition nor confirmed by the Conjecture; either because system (3) admits

no solution on minimal clique covers or because even for such covers exhaustively checking for the absence of a solution is not computationally feasible. We present the case-by-case analysis of the 9 graphs in Appendix A.

As the number of vertices considered increases, we expect the number of graphs left undecided by Proposition 2.2 as well as their diversity to increase quickly. This explains why we could not determine the number of realizable graphs on 8 vertices. Indeed, in all cases unconcerned by Proposition 2.2 (that represents 1662 graphs on 8 vertices, of which 23 are realizable trees) for which no solution to system (3) can be found more ad-hoc arguments must be devised. The number of graphs in this situation is much larger than 9, not to mention that the arguments are both more difficult to find and more diverse than for the 9 graphs mentioned above.

We conclude this section by an observation about those graphs that are unrealizable. Since they are not, we might think that the elements of the corresponding trace monoids \mathcal{T} cannot be drawn as cycles on a digraph and are therefore essentially different from walks and hikes. This however is incorrect.

Proposition 3.7. Let H be a graph that is unrealizable. Then there exist at least one realizable graph H' such that H is an induced subgraph of H' . Equivalently, let \mathcal{T} be a trace monoid that is not a hike monoid. Then \mathcal{T} is a submonoid of a hike monoid \mathcal{H}' .

This indicates that elements of \mathcal{T} are walks and walk-like objects after all, yet these are not drawable as such by themselves. Perhaps more strikingly, since \mathcal{T} is a monoid in its own right it is *algebraically closed*. Elements of $\mathcal{H}' \setminus \mathcal{T}$ are, in this sense, algebraically unrelated to those of \mathcal{T} . Yet allowing these additional elements turns the undrawable members of \mathcal{T} into drawable objects.

Proof. Given any graph H , we can always draw simple cycles sharing vertices so as to produce the cycle dependencies dictated by H . Since H is unrealizable this procedure must have created additional simple cycles in the graph G' just drawn. This implies that H is an induced subgraph of $H' = \phi(G')$. \square

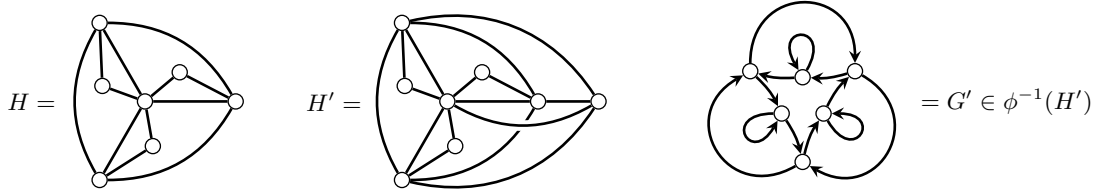


Figure 3: An example of Proposition 3.7, graph H is unrealizable yet appears as an induced subgraph of H' which is realized by the digraph G' shown on the right.

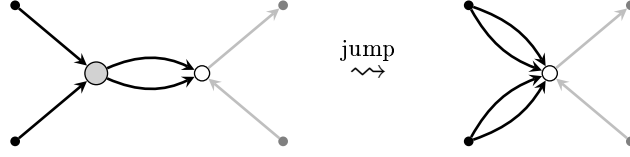
Remark 3. Proposition 3.7 generalizes our preceding observation in (2) about the unrealizable square graph to all unrealizable graphs. As a corollary there cannot be a forbidden induced-subgraph criterion for realizability.

4. Injectivity: graphs with isomorphic hike monoids

4.1. Transformations between ϕ -equivalent graphs

We now turn to question 2 of §2, that is we seek to find a description of ϕ -equivalence. We first present two sufficient conditions for two digraphs to be ϕ -equivalent, then show how these fail to be necessary on examples we found to be particularly telling of the difficulty of describing ϕ -equivalence classes.

We start with three digraph transformations which preserve the hike monoid. Let $G = (V, E)$ be a digraph. For (v_1, v_2) , an edge of G , we call *reversing* (v_1, v_2) the transformation consisting in replacing the edge (v_1, v_2) by the edge (v_2, v_1) . For v a vertex of G , we call *jumping* the vertex v the transformation which consists in removing v from V and adding an edge (v_1, v_2) for each pair of edges (v_1, v) and (v, v_2) . See below for an example of jumping at the gray vertex:



Proposition 4.1. Let G be a digraph.

1. Reversing all the edges of G preserves the hike monoid of G .
2. Let v_1, v_2 be two vertices of G and all outgoing edges from v_1 point to v_2 . Then jumping v_1 preserves the hike monoid of G .
3. Let v_1, v_2 be two vertices of G and all in-going edges to v_2 come from v_1 . Then jumping v_2 preserves the hike monoid of G .

Proof.

1. Let G' be the digraph obtained by reversing all the edges of G . Then there is a simple cycle (v_1, \dots, v_n) in G' if and only if there is a simple cycle (v_n, \dots, v_1) in G , hence $G \sim_\phi G'$.
2. Let G' be the digraph obtained by jumping v_1 . Since the only outgoing edge from v_1 are of the form (v_1, v_2) , every simple cycle of G passing by v_1 must also pass by v_2 . Hence there is a bijection between the simple cycles of G and G' which sends simple cycles without v_1 on themselves and simple cycles with v_1 on the simple cycle obtained by removing v_1 . Hence $G \sim_\phi G'$.
3. The last item is a direct consequences of the previous two.

□

Proposition 4.1 permits a reduction of the problem of realizability to cubic graphs, that is digraphs where all vertices have total degree 3:

Corollary 4.2. A graph is realizable if and only if it is realizable by a cubic graph.

Proof. Let H be a connected realizable graph and let $G = (V, E)$ be a strongly connected digraph realizing H and v be a vertex of G . Since G is strongly connected, v can not have a total degree $d(v)$ equal to 1 and $d(v) = 2$ if and only if v has one in-going edge and one out-going edge. In this case, from Proposition 4.1, we know that we can jump v while conserving H . Suppose now $d(v) > 3$ and respectively denote by u_0, \dots, u_{n-1} and v_1, \dots, v_k the vertices in the sets $\{u \in V \mid (u, v) \in E\}$ and $\{u \in V \mid (v, u) \in E\}$. We define the digraph G' by transforming G as follow (see Figure 4 for an example):

- remove the vertex v from G ,
- add $n + k - 2$ vertices w_1, \dots, w_{n+k-2} and the edges (w_i, w_{i+1}) for $1 \leq i \leq n + k - 3$,
- add edges (u_0, w_1) , (u_i, w_i) for $1 \leq i \leq n - 1$, (w_{n+i-1}, v_i) for $1 \leq i \leq k - 1$ and (w_{n+k-2}, v_k) .

By construction, vertices w_1, \dots, w_{n+k-2} have total degree 3 and we can recover digraph G by jumping them. Since by Proposition 4.1 this preserves the hike monoid, G' also realizes H . By applying the same transformation to every vertices of G , we obtain a cubic graph. □

Proposition 4.1 give us a potential characterisations of ϕ -equivalence: $G_1 \sim_\phi G_2$ if and only if G_1 and G_2 reduce to the same digraph after recursively applying the transformations of Proposition 4.1 until we obtain a digraph where each vertex has in-degree and out-degree at least 2. The digraph so obtained would then be a canonical representative of the ϕ -equivalent class $\phi^{-1}(H)$ to which G_1 and G_2 belong. An alternative canonical representative would be a cubic graph realizing

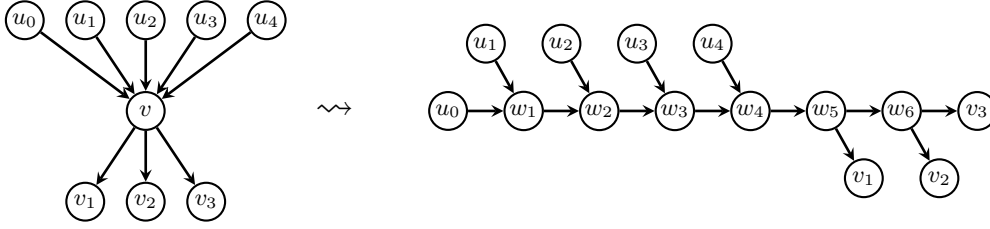
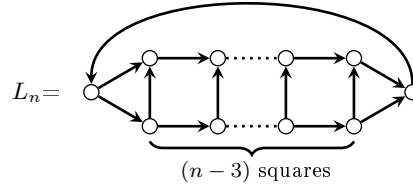


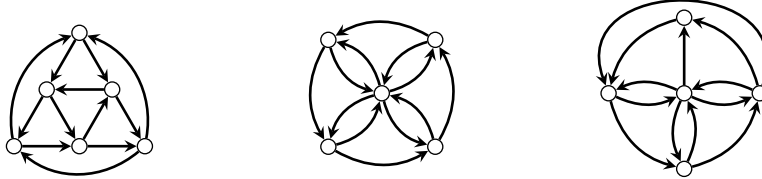
Figure 4: How to transform vertices and their adjacent edges so as to form a cubic graph: an illustration of Corollary 4.2 with $n = 4$ and $k = 3$.

H , the existence of which is guaranteed by Corollary 4.2. For example, all complete graphs on n vertices $H = K_n$ are realizable, being realized by the bouquet graph B_n with n self-loops on a single vertex, itself irreducible under Proposition 4.1. Alternatively, the cubic ladder digraph L_n

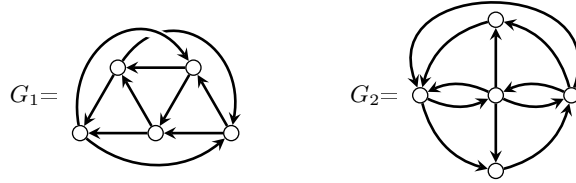


satisfies $\phi(L_n) = K_n$ for $n \geq 3$, and L_n reduces to B_n under Proposition 4.1.

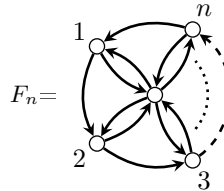
Unfortunately while it is true that two digraphs reducing to the same digraph are necessarily ϕ -equivalent, the converse proposition does not hold. For example the following three digraphs are realizations of the complete graph on 17 vertices K_{17} , yet do not reduce to the same digraph under the transformations listed above



More generally we found that for most graph theoretic properties, we could construct a realization of K_n that has the property and another that does not have it. For example both of the following digraphs realize K_{12}



yet G_1 is vertex-transitive while G_2 is not, G_2 is planar and G_1 is not; or consider the flower graph F_n on n petals



for which $\phi(F_n) = K_{n^2+1}$. Flower F_n has a highly skewed degree distribution while that of the ladder L_{n^2+1} is uniform; etc. This means that even considering the purportedly simpler case of the complete graphs $H = K_n$, we still find a profusion of ϕ -equivalent graphs unaccounted for by Proposition 4.1 which moreover do not seem to be related to one another in any obvious way.

These observations show that to pursue a full characterisation of ϕ -equivalence we need an exhaustive description of the hike dependency graphs of all graphs with in- and out-degrees at least 2. We could not find such a description—and a plethora of cases make this task clearly daunting. For a realizable graph H we do however have a lower bound and two upper bounds on the number of vertices and edges that are satisfied by at least one graph in $\phi^{-1}(H)$ which provide some constraint on $\phi^{-1}(H)$. To describe these bounds consider $G = (V_G, E_G)$ a strongly connected digraph with in- and out-degrees at least 2 and $H = (V_H, E_H) = \phi(G)$.

First, as remarked in the previous section, every vertex of G correspond to a clique in H , consequently $|V_G|$ must be at least the size of a minimal clique cover of H .

Second, to give upper bounds on $|V_G|$ and $|E_G|$, we begin by briefly recalling the notion of ear decomposition of a graph while we refer the reader to [17] for a thorough discussion of this topic. An *ear decomposition* of G is a sequence (G_1, \dots, G_n) of directed subgraphs of G , called *ears*, such that:

- G_1 is a simple cycle,
- for $i > 1$, G_i is a path (possibly closed) with exactly its start and end vertices already present among the vertices of G_j for $1 \leq j < i$,
- $G = \bigcup G_i$.

Consider the procedure of constructing G by starting with $\tilde{G}_1 = G_1$ and recursively constructing \tilde{G}_i as $\tilde{G}_i = \tilde{G}_{i-1} \cup G_i$ for $i > 1$. Since the first ear has as many vertices as edges and that the addition of every other ear adds exactly one more edge than vertices, the length of an ear decomposition is always equal to $|E_G| - |V_G| + 1$. Now remark that graphs \tilde{G}_i are all strongly connected so that adding G_{i+1} to \tilde{G}_i forms at least one additional cycle. Hence we have the inequality $|V_H| \geq |E_G| - |V_G| + 1$. Since furthermore we can always reduce G to have in- and out-degrees at least 2, we have that $|E_G| = \frac{1}{2} \sum_{v \in V_G} d_{in}(v) + d_{out}(v) \geq \frac{4}{2} |V_G| = 2|V_G|$ giving the upper bounds $|V_H| - 1 \geq |V_G|$ and $2|V_H| - 2 \geq |E_G|$.

4.2. Implications of sharing the same dependency graph

4.2.1. Simple graphs do not relate well with their simple cycles

Let H be a realizable graph. As we have seen so far, digraphs realizing H may be very diverse, not even sharing basic graph theoretic properties such as regularity. By contrast, in this section we present some properties they must necessarily share by virtue of the isomorphism between their hike monoids. The first result on this matter was obtained in [10] by regarding graphs as bidirected digraphs.

Theorem 4.3 (Giscard and Rochet, 2016 [10]). Let G be a connected graph with no self-loop. Then the hike dependency graph H of G determines G uniquely up to isomorphism, unless $G \in \{K_3, K_{1,5}\}$.

While this result seems to be quite powerful at first glance, the seemingly harmless assumption that the graph is bidirected (i.e., effectively undirected) is essential. Should even a single edge be directed the theorem would completely fail to hold. Indeed, the proof relies exclusively on simple cycles of length 2 (called backtracks) sustained by every undirected edge $\{i, j\}$ on G . Vertices of H corresponding to backtracks in G can always be identified even though the lengths of the simple cycles corresponding to the vertices of H are not known (with the single exception $H = K_5$ leading to the $\{K_3, K_{1,5}\}$ special case). Now the induced subgraph of H formed by the backtracks is the line graph $L(G)$ of G , which is well known to determine G uniquely. In other terms, the theorem makes no use of longer simple cycles, so much so that if the information regarding the backtracks is discarded from H , that is we only have access to $\phi(G) \setminus L(G)$, then graphs suffer from exactly the same shortfalls as directed graphs when it comes to relating them to their closed walks. That is,

- There are graphs G and G' with $\phi(G)/L(G) = \phi(G')/L(G')$ and G and G' do not share fundamental graph theoretic properties. See Figure 5 below.

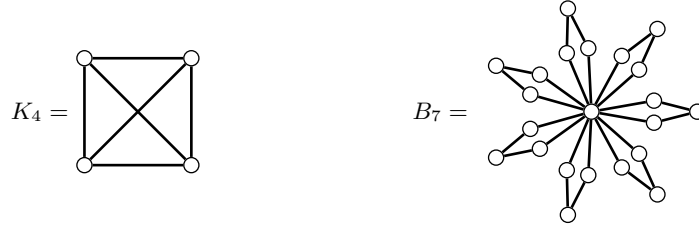


Figure 5: A pair of graphs with $\phi(K_4)/L(K_4) = \phi(B_7)/L(B_7)$, graph K_4 on the left is Hamiltonian and vertex-transitive but not bipartite unlike the bouquet graph B_7 on the right, which is bipartite and neither Hamiltonian nor vertex-transitive.

- Given a graph H , deciding whether there exists a graph G such that $H = \phi(G)/L(G)$ is widely open.

These issues are pressing since many tools of modern network analysis and algebraic graph theory explicitly disregard or forbid backtracks. This includes backtrackless walks methods [18], Ihara zeta function and primitive orbits [19] and open problems about graphs and cycles, foremost among which is the cycle double cover conjecture [20].

4.2.2. Invariants of hike monoids

Recall from §1.1.2 that for a digraph G , $H = \phi(G)$ is the dependency graph of its hike monoid, which is the trace monoid generated by its simple cycles under the partial commutation rule that two cycles commute if and only if they share no vertex.

Given that isomorphisms of hike monoids implement profound changes on digraphs, it is worth investigating quantities left invariant by these isomorphisms. The first type of such invariants are algebraic quantities based on the adjacency matrix.

Proposition 4.4. Let G and G' be two ϕ -equivalent digraphs and let \mathbf{A} and \mathbf{A}' be their adjacency matrices, respectively. Then:

- 1) $\det(\mathbf{I} - \mathbf{A}) = \det(\mathbf{I}' - \mathbf{A}')$,
- 2) $\text{perm}(\mathbf{I} + \mathbf{A}) = \text{perm}(\mathbf{I}' + \mathbf{A}')$,

where \mathbf{I} and \mathbf{I}' are identity matrices of appropriate sizes.

Proof. A self-avoiding hike is a hike $h = c_1 \cdots c_k$ in which all simple cycles c_j are vertex-disjoint, hence commute. Consequently, a hike is self-avoiding if and only if the set $\{c_1, \dots, c_k\}$ forms a clique in the complement graph H^c of the hike dependency graph. It follows that there is a bijection ψ between the sets of self-avoiding hikes on two ϕ -equivalent digraphs that preserves the number of simple cycles in each self-avoiding hike. From the definition of the matrix determinant in terms of permutations and the representation of permutations as disjoint cycles we have

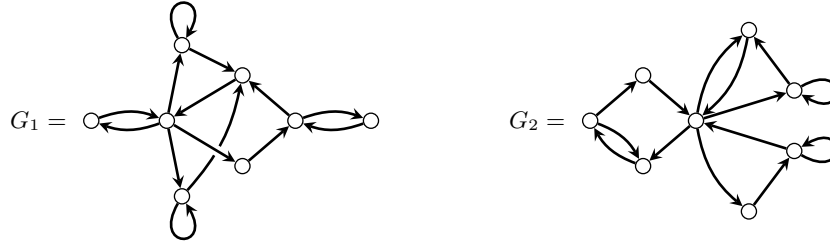
$$\sum_{h \text{ s.a.}} (-1)^{\Omega(h)} = \det(\mathbf{I} - \mathbf{A}),$$

where the sum runs over self-avoiding hikes, $\Omega(h)$ denotes the number of simple cycles in h and thus $(-1)^{\Omega(h)}$ is the signature of the permutation with cycle decomposition h (see [8, 7]). The sum is left unchanged by the monoid isomorphism ψ , which entails assertion 1) Indeed, self-avoiding hikes are cliques in the independency graph H^c which is invariant under ψ so both the sizes of these cliques (the number $\Omega(h)$) as well as their total number is preserved by ψ . This last observation implies assertion 2) as $\sum_{h \text{ s.a.}} 1 = \text{perm}(\mathbf{I} + \mathbf{A})$. \square

Remark 4. The list of algebraic invariants provided in Proposition 4.4 is *not* exhaustive, many more invariants can for example be inferred from [21]. To see this consider two algebraic invariants of hike monoids $a(G)$ and $b(G)$ defined on a digraph G and all its induced subgraphs $g \prec G$. Then

the induced subgraph convolution $(a * b)(G) = \sum_{g \prec G} a(g)b(G \setminus g)$ is also an algebraic invariant of hike monoids. In other terms if G and G' are two ϕ -equivalent digraphs, then $a(G) = a(G')$, $b(G) = b(G')$ and $(a * b)(G) = (a * b)(G')$. As an instance of this we unify and generalise assertions 1) and 2) with $\text{perm}(\mathbf{I} + \mathbf{A})^{*k}$, the k th-induced subgraph convolution of the permanent with itself, which is therefore an invariant of hike monoids for all $k \in \mathbb{Z}$ (the Hopf algebraic relation $\text{perm}(\mathbf{I} + \mathbf{A})^{*-1} = \det(\mathbf{I} + \mathbf{A})$ gives meaning to the negative $*$ powers of the permanent [21]). We may verify this claim directly by writing [21] $\text{perm}(\mathbf{I} + \mathbf{A})^{*k} = \sum_{h \text{ s.a.}} k^{\Omega(h)}$ and reasoning as in the proof of Proposition 4.4.

If the isomorphism of hike monoids between \mathcal{H}_G and $\mathcal{H}_{G'}$ is length preserving, then all algebraic quantities computable from their adjacency matrices are identical (e.g. degree sequence, graph spectra, permanental polynomial $\text{perm}(\mathbf{I} - z\mathbf{A})$ etc). Allowing for the addition of transient vertices it is always possible to construct a pair of digraphs related by a length-preserving isomorphism of hike monoids from a pair of digraphs related by an isomorphism that does not preserve the length. This gives a systematic mean of generating cospectral pairs, although the digraphs so generated are much more similar than just cospectral (remark also that there are pairs of cospectral graphs do not share the same hike dependency graph). Below is an example of digraphs related by such a morphism:



The second type of invariant are combinatorial invariants sensitive to how walks and hikes are composed of cycles:

Proposition 4.5. Let $G = (V_G, E_G)$ be a digraph, \mathcal{H}_G its hike monoid and $\mathcal{W}_G \subset \mathcal{H}_G$ be the set of walks in \mathcal{H}_G , that is the set of hikes $h = c_1 \cdots c_k \in \mathcal{H}_G$ with a unique right simple cycle c_k . Let $\Omega(h)$ be the number of simple cycles in a hike h . Let $f_{\mathcal{H}_G}^\Omega(z) := \sum_{h \in \mathcal{H}_G} z^{\Omega(h)}$, $f_{\mathcal{W}_G}^\Omega(z) := \sum_{h \in \mathcal{W}_G} z^{\Omega(h)}$, be the associated ordinary generating functions on hikes and walks, respectively.

Then $G \sim_\Phi G'$ implies

1. $f_{\mathcal{H}_G}^\Omega(z) = f_{\mathcal{H}_{G'}}^\Omega(z)$;
2. $f_{\mathcal{W}_G}^\Omega(z) = f_{\mathcal{W}_{G'}}^\Omega(z)$.

Now consider the set $\mathcal{W}_{G:i \rightarrow j}$ of rooted walks w from vertex i to vertex j in G . Let $\Omega(w)$ designate the number of loops erased from w following Lawler's loop erasing procedure [22]. Let $\sigma(\cdot)$ be an additive function on hikes. Then $G \sim_\Phi G'$ implies that for all $i, j \in V_G$ there exists $i', j' \in V_{G'}$ with

3. $f_{\mathcal{W}_{G:i \rightarrow j}}^\Omega(z) = f_{\mathcal{W}_{G':i' \rightarrow j'}}^\Omega(z)$;
4. The shape of the branched continued fraction representation of the generating series associated with an additive function σ defined on rooted walks from i to j on G is the same as that representing the generating function of σ on rooted walks from i' to j' on G' .

Proof. Claims 1. and 2. follow immediately from the definitions of $f_{\mathcal{H}_G}^\Omega(z)$ and $f_{\mathcal{W}_G}^\Omega(z)$ as sums over monoid elements. Since \mathcal{H}_G and $\mathcal{H}_{G'}$ are isomorphic, these functions are the same on G and G' provided $\Omega(\cdot)$ and $\omega(\cdot)$ take on the same value over a hike in \mathcal{H}_G and its counterpart under isomorphism in $\mathcal{H}_{G'}$, which is clearly true.

For assertions 3 and 4, consider $\sigma : \mathcal{W}_G \rightarrow \mathbb{R}$ an additive function on walks, i.e., for $w \in \mathcal{W}_G$ with $w = c_1 \cdots c_k$ we have $\sigma(w) = \sum_{i=1}^k \sigma(c_i)$. Function σ extends naturally to rooted walks so we can consider generating function $f_{\mathcal{W}_{G:i \rightarrow j}}^\sigma(z) := \sum_{h \in \mathcal{W}_{G:i \rightarrow j}} z^{\sigma(h)}$. Any such sum over all rooted walks has a unique representation as a branched continued fraction of finite depth and breadth called a path-sum [23]. Each term of a path-sum corresponds to a rooted simple cycle on G . The continued fraction gains a branch if two or more rooted simple cycles share the same root. A branch gains depth when a simple cycle's root is located inside of another rooted simple cycle. A branch gains breadth when several simple cycles' roots are distinct and located inside of another simple cycle. Overall this implies that the shape of the path-sum is that of the spanning tree of H whose level one vertices are the simple cycles crossing vertex i . Consequently, this shape is an invariant of isomorphisms of hike monoids since they preserve H . Assertion 3 follows from the peculiar choice $\sigma = \Omega$ and $\Omega(c) = 1$ for any simple cycle $c \in \mathcal{C}$. \square

Example 4.6. Consider the following two graphs with identical hike dependency graph:

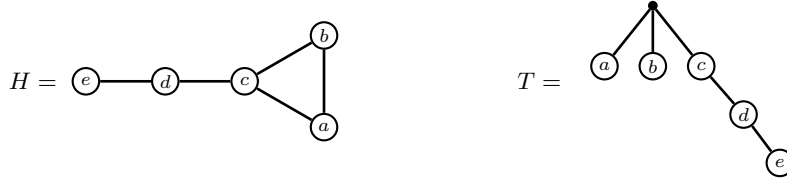


Let σ be the length function ℓ , which is additive as necessary. Then the path-sum continued fraction representation of the ordinary generating function of all rooted closed walks from vertex \circ to itself on G and G' associated with the length function is

$$\sum_{w \in \mathcal{W}_{G:\circ \rightarrow \circ}} z^{\ell(w)} = \frac{1}{1 - z - z^2 - \frac{z^3}{1 - \frac{z^2}{1 - z}}} = (\mathbf{I} - z\mathbf{A}_G)_{\circ\circ}^{-1},$$

$$\sum_{w \in \mathcal{W}_{G':\circ \rightarrow \circ}} z^{\ell(w)} = \frac{1}{1 - z - z - \frac{z^2}{1 - \frac{z^4}{1 - z^2}}} = (\mathbf{I}' - z\mathbf{A}_{G'})_{\circ\circ}^{-1},$$

where \mathbf{A}_G and $\mathbf{A}_{G'}$ are the adjacency matrices of G and G' , respectively. The common structure T of both fractions is readily apparent and stems from the hike dependency graph $H = \phi(G) = \phi(G')$ as explained in the proof of Proposition 4.5:



In G , a and e are self-loops, b and d are backtracks and c is a triangle. In G' , a and b are self-loops, c and e are backtracks and d is a square. Since Ω is additive the same structure T appears in $f_{\mathcal{W}_{G:1 \rightarrow 1}}^\Omega(z) = f_{\mathcal{W}_{G':1 \rightarrow 1}}^\Omega(z)$,

$$f_{\mathcal{W}_{G:1 \rightarrow 1}}^\Omega(z) = f_{\mathcal{W}_{G':1 \rightarrow 1}}^\Omega(z) = \frac{1}{1 - z - z - \frac{z}{1 - \frac{z}{1 - z}}}.$$

5. Conclusion

In this work we investigated the relation between digraphs and their walks and found it to be rather weak: no graph theoretic property seems to be directly related to the arrangement of simple cycles on a digraph. Among the graph properties we found may be lost whilst leaving the hike monoidal structure of cycles invariant are vertex-transitivity, regularity, planarity, bipartiteness, (bi)directedness, Hamiltonicity, being Eulerian, being chordal, being triangle-free,

chromatic number, graph spectra, in- and out-degree distributions and a majority of algebraic quantities computable from adjacency matrices. This list is undoubtedly non-exhaustive. As a corollary, there is a great variety of digraphs with isomorphic hike monoids and characterizing all transformations relating such digraphs remains completely open.

Conversely, just deciding which arrangements of simple cycles exist at all is highly non-trivial, even allowing for multidigraphs to realize them as was done here. We have shown that realizability is equivalent to the existence of integer solutions to polynomial systems of equations, making realizability decidable but no less obscure. We emphasize that both the characterization and existence questions concern simple undirected graphs too as soon as information about their line graph is removed. That is, simple graphs do not relate to their simple cycles of length $\ell \neq 2$ any better than multidigraphs do to their simple cycles of any length.

All of this demonstrates that walks are in fact much less dependent on the digraphs on which they take place than might have first been thought; and that a “*theory of walks*” distinct from graph theory needs to be developed. Here we proceeded by relying on hike monoids, which provide a representation for walks and walk-like objects that is markedly detached from graphs sustaining those walks. As monoids, hike monoids are plain trace monoids. Unfortunately, while the latter are well understood, there is no simple way to know which trace monoids are hike monoids. Although we have formulated most of our results in terms of dependency graphs, expressing them directly as statements on trace monoids makes this fact even more clear. Consider for example the following family of monoids with identical independence relations but differing number of generators:

$$\begin{aligned} T_0 &= \{a, b, c, d \mid ac = ca, bd = db\}, \\ T_n &= \{a, b, c, d, x_1, \dots, x_n \mid ac = ca, bd = db\}. \end{aligned}$$

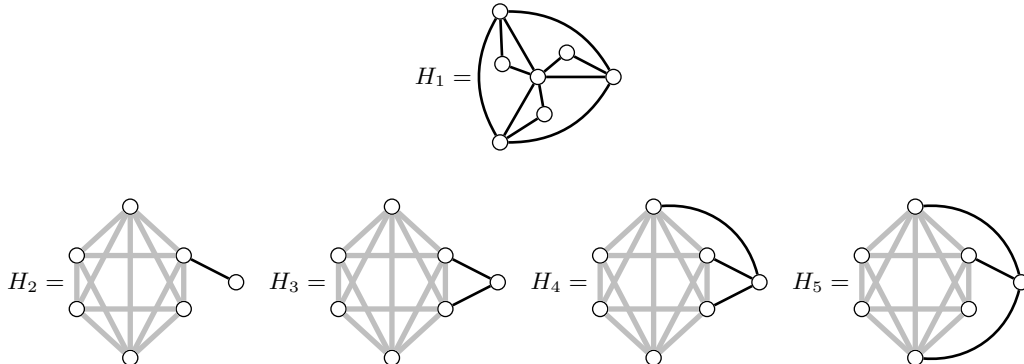
Among these very similar trace monoids, we know that $T_0, T_1, T_3, T_5, T_9, T_{17}$ and T_{29} are not hike monoids, while all others are. Trace theory does not seem to be any better equipped than graph theory to address the questions raised here.

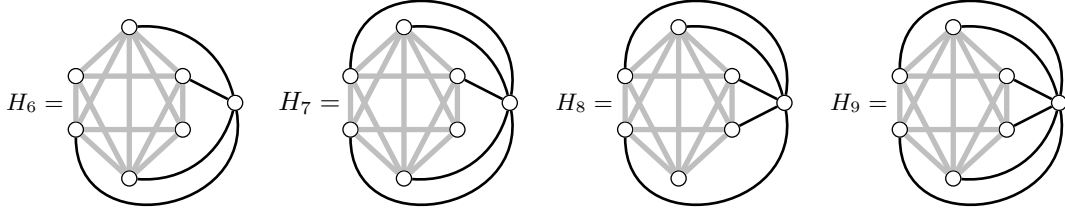
Acknowledgments

T. K. acknowledges postdoctoral funding by the Université du Littoral Côte d’Opale. This work is supported by the French National Research Agency (ANR) through the ANR-19-CE40-0006 project Algebraic Combinatorics of Hikes On Lattices (Alcohol).

AppendixA. Nine unrealizable graphs on 7 vertices

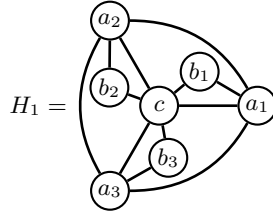
Consider the following nine graphs on seven vertices





System (3) admits no solution on minimal clique covers in the cases of $H_1, H_2, H_3, H_4, H_7, H_8$ and H_9 , while exhaustively checking for the absence of such a solution is not computationally feasible for H_5 and H_6 . We present two ad-hoc arguments deciding the unrealizability of these graphs: the first for H_1 and the second for all other graphs.

H_1 : Let us label H_1 as follows:



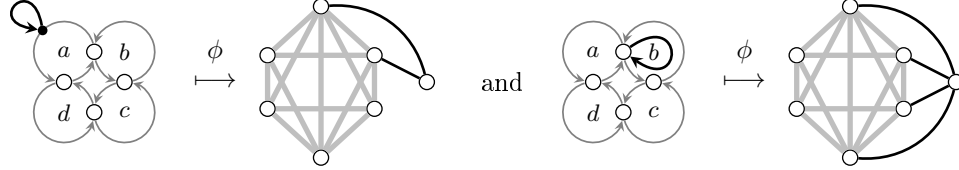
Suppose that H_1 is realizable and let G be a digraph realizing it. Each triangle $(ca_1b_1, ca_1a_2, \dots)$ in H_1 corresponds to three simple cycles in G sharing a common vertex: indeed, if there was no such vertex then by Proposition 2.2 there would be two vertices in H_1 linked to every vertex of the triangle, which contradicts the structure of H_1 .

Suppose now that there is no vertex common to all 4 cycles a_1, a_2, a_3 and c . Denote by v_1 a vertex common to a_1, a_2 and c ; by v_2 a vertex common to a_2, a_3 and c ; and by v_3 a vertex common to a_3, a_1 and c . For $i \in \{1, 2, 3\}$, and v, v' two vertices of a_i , denote by $v \rightarrow_i v'$ the path from v to v' in a_i . Let us now consider cycles $v_1 \rightarrow_2 v_2 \rightarrow_3 v_3 \rightarrow_1 v_1$ and $v_3 \rightarrow_3 v_2 \rightarrow_2 v_1 \rightarrow_1 v_3$ and show that they are simple. Suppose that the first of these two cycle is not simple and, without loss of generality, let v be a vertex such that $v_1 \rightarrow_2 v_2 = v_1 \rightarrow_2 v \rightarrow_2 v_2$, $v_2 \rightarrow_3 v_3 = v_2 \rightarrow_3 v \rightarrow_3 v_3$ and $v \rightarrow_2 v_2 \rightarrow_3 v$ is a simple cycle. Then, by construction, this simple cycle can neither be any of a_1, a_2, a_3 and c since there are missing vertices; nor can it be any of the b_i s since it is in contact with two a_i s which would contradicts the structure of H_1 . Hence $v_1 \rightarrow_2 v_2 \rightarrow_3 v_3 \rightarrow_1 v_1$ is a simple cycle and, similarly, so is $v_3 \rightarrow_3 v_2 \rightarrow_2 v_1 \rightarrow_1 v_3$. These two simple cycles can not be identifiable with one of the a_i s since else one of the vertices v_i would be common to a_1, a_2, a_3 and c . Neither can these two cycles be identifiable with one of the b_i s because they are in contact with more than two a_i s. Hence they must be distinct yet both be equal to c , a contradiction.

We are thus forced to consider that there exists a vertex v_0 common to all 4 cycles a_1, a_2, a_3 and c . As earlier, denote by v_1 a vertex common to a_1, b_1 and c ; by v_2 a vertex common to a_2, b_2 and c ; and by v_3 a vertex common to a_3, b_3 and c . Then $\{v_0, v_1, v_2, v_3\}$ corresponds to a minimal clique cover of H_1 , implying that G would provide a solution to system (3) with a minimal clique cover which, we have checked, does not exist.

H_2 to H_9 : We know from Proposition 2.2 and the example preceding this proposition that if a realizable graph has a square as induced subgraph then necessarily this square is part of an induced subgraph with the shape of H on the left of 2. Furthermore, we also know that the simple cycles corresponding to the vertices of this subgraph must be organised as in Figure 2. Now remark that adding an edge which is not a self-loop to Figure 2 always creates two simple cycles so that the corresponding image by ϕ must have at least 8 vertices. Consequently, the only way to realize a graph with 7 vertices and a square as induced subgraph is to add

a self-loop somewhere in Figure 2. There are only two ways to do so which yield distinct hike monoids: either by adding the self-loop on a vertex that lies inside of one of the simple cycles a, b, c, d and in no other cycle; or on a vertex that is in common to two cycles. The image by ϕ of the digraphs so-obtained are respectively:



Since graphs H_2 to H_9 all have a square as induced subgraph but are different from the above two graphs, they are unrealizable.

References

- [1] X. Qi, E. Fuller, Q. Wu, Y. Wu, C.-Q. Zhang, Laplacian centrality: A new centrality measure for weighted networks, *Information Sciences* 194 (2012) 240–253, intelligent Knowledge-Based Models and Methodologies for Complex Information Systems. doi:<https://doi.org/10.1016/j.ins.2011.12.027>. URL <https://www.sciencedirect.com/science/article/pii/S0020025511006761>
- [2] T. Opsahl, F. Agneessens, J. Skvoretz, Node centrality in weighted networks: Generalizing degree and shortest paths, *Social Networks* 32 (3) (2010) 245–251. doi:<https://doi.org/10.1016/j.socnet.2010.03.006>. URL <https://www.sciencedirect.com/science/article/pii/S0378873310000183>
- [3] E. Estrada, J. Rodríguez-Velázquez, Subgraph centrality in complex graphs, *Physical Review E* 5 (2005) 056103.
- [4] J. Ramon, T. Gärtner, Expressivity versus efficiency of graph kernels, in: L. D. Raedt, T. Washio (Eds.), *Proceedings of the First International Workshop on Mining Graphs, Trees and Sequences (MGTS 2003) at the 14th European Conference on Machine Learning and 7th European Conference on Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD 2003)*, September 22 and 23, 2003, Cavtat-Dubrovnik, Croatia, 2003, pp. 65–74.
- [5] P. Mahé, J.-P. Vert, Graph kernels based on tree patterns for molecules, *Machine Learning* 75 (1) (2009) 3–35. doi:[10.1007/s10994-008-5086-2](https://doi.org/10.1007/s10994-008-5086-2). URL <https://doi.org/10.1007/s10994-008-5086-2>
- [6] N. M. Kriege, F. D. Johansson, C. Morris, A survey on graph kernels, *Applied Network Science* 5 (1) (2020) 6. doi:[10.1007/s41109-019-0195-3](https://doi.org/10.1007/s41109-019-0195-3). URL <https://doi.org/10.1007/s41109-019-0195-3>
- [7] P. Cartier, D. Foata, *Problèmes combinatoires de commutation et réarrangements*, Lecture notes in mathematics 85.
- [8] G. X. Viennot, Heaps of pieces, i: Basic definitions and combinatorial lemmas, *Annals of the New York Academy of Sciences* 576 (1) (1989) 542–570.
- [9] C. Krattenthaler, The theory of heaps and the Cartier–Foata monoid, Appendix of the electronic edition of “Problèmes combinatoires de commutation et réarrangements”.
- [10] P.-L. Giscard, P. Rochet, Algebraic combinatorics on trace monoids: Extending number theory to walks on graphs, *SIAM Journal on Discrete Mathematics* 31 (2) (2017) 1428–1453. arXiv:<https://doi.org/10.1137/15M1054535>, doi:[10.1137/15M1054535](https://doi.org/10.1137/15M1054535). URL <https://doi.org/10.1137/15M1054535>

- [11] M. Mosbah, N. Saheb, A syntactic approach to random walks on graphs, in: R. H. Möhring (Ed.), *Graph-Theoretic Concepts in Computer Science*, Springer Berlin Heidelberg, Berlin, Heidelberg, 1997, pp. 258–272.
- [12] M. Mosbah, N. Saheb, Formal rational fractions and random walks on cycles, *Discrete Mathematics* 217 (1) (2000) 301–313. doi:[https://doi.org/10.1016/S0012-365X\(99\)00269-1](https://doi.org/10.1016/S0012-365X(99)00269-1). URL <https://www.sciencedirect.com/science/article/pii/S0012365X99002691>
- [13] S. Thwaite, A family of partitions of the set of walks on a directed graph (2014). doi:10.48550/ARXIV.1409.3555. URL <https://arxiv.org/abs/1409.3555>
- [14] C. Lindorfer, W. Woess, The language of self-avoiding walks, *Combinatorica* 40 (5) (2020) 691–720. doi:10.1007/s00493-020-4184-z. URL <https://doi.org/10.1007/s00493-020-4184-z>
- [15] P. A. MacMahon, *Combinatory analysis*, Cambridge: The University Press, 1915.
- [16] B. D. McKay, A. Piperno, Practical graph isomorphism, ii, *Journal of Symbolic Computation* 60 (2014) 94–112. doi:<https://doi.org/10.1016/j.jsc.2013.09.003>. URL <https://www.sciencedirect.com/science/article/pii/S0747717113001193>
- [17] J. Bang-Jensen, G. Gutin, *Digraphs Theory, Algorithms and Applications*, Springer-Verlag, 2007, Ch. 7.2.
- [18] F. Aziz, R. C. Wilson, E. R. Hancock, Backtrackless walks on a graph, *IEEE Transactions on Neural Networks and Learning Systems* 24 (6) (2013) 977–989. doi:10.1109/TNNLS.2013.2248093.
- [19] A. Terras, *Zeta Functions of Graphs: A Stroll through the Garden*, 1st Edition, Cambridge University Press, Cambridge, 2011.
- [20] F. Jaeger, A survey of the cycle double cover conjecture, in: B. Alspach, C. Godsil (Eds.), *Annals of Discrete Mathematics* (27): Cycles in Graphs, Vol. 115 of North-Holland Mathematics Studies, North-Holland, 1985, pp. 1–12. doi:[https://doi.org/10.1016/S0304-0208\(08\)72993-1](https://doi.org/10.1016/S0304-0208(08)72993-1). URL <https://www.sciencedirect.com/science/article/pii/S0304020808729931>
- [21] P.-L. Giscard, P. Rochet, R. C. Wilson, A Hopf algebra for counting cycles, *Discret. Math.* 341 (2018) 1439–1448.
- [22] G. F. Lawler, Loop-erased self-avoiding random walk and the laplacian random walk, *Journal of Physics A: Mathematical and General* 20 (13) (1987) 4565.
- [23] P.-L. Giscard, S. J. Thwaite, D. Jaksch, Continued fractions and unique factorization on digraphs, arXiv preprint, arXiv:1202.5523.