



The no-meet matroid

Walid Ben-Ameur, Natalia Kushik, Alessandro Maddaloni, José Neto, Dimitri Watel

► To cite this version:

Walid Ben-Ameur, Natalia Kushik, Alessandro Maddaloni, José Neto, Dimitri Watel. The no-meet matroid. Discrete Applied Mathematics, 2022, 10.1016/j.dam.2022.12.008 . hal-03955928

HAL Id: hal-03955928

<https://hal.science/hal-03955928>

Submitted on 25 Jan 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.

The No-Meet Matroid

Walid Ben-Ameur¹, Natalia Kushik¹, Alessandro Maddaloni¹, José Neto¹, and Dimitri Watel²

¹Samovar, Telecom SudParis, Institut Polytechnique de Paris,
Palaiseau, France

²Samovar, ENSIIE, Evry, France

November 17, 2022

Abstract

Given a digraph and a subset of vertices representing initial positions, we study the existence problem of infinitely long walks, one starting from each initial position, that never meet. We show that the subsets for which this is possible constitute the independent sets of a matroid. We prove that the independence oracle is polynomial-time. We also provide a more efficient algorithm to compute the size of a basis. Then we focus on the orientation problem of the undirected edges of a mixed graph to either maximize or minimize the rank of a subset. Some NP-hardness results and inapproximability results are proved in the general case. Polynomial-time algorithms are described for subsets of size 1. We also report several connections with other matroid classes. For example, we show that the class of no-meet matroids strictly contains transversal matroids while it is strictly contained inside gammoids. Some extensions and related applications are also discussed.

Keywords: Matroids, Graphs, Cycles, Orientation, Complexity, PCP

1 Introduction

Let $G = (V, E)$ denote a digraph (directed graph) with vertex set V and edge set E . The graph G may contain loops. Let $S \subseteq V$. A vertex subset S' is called a *successor* of S if S' can be obtained from S by replacing each vertex in S by one of its successors. And S' is called *valid successor* of S if $|S'| = |S|$. The set of all valid successors of S is denoted by $\mathcal{M}(S)$.

Formally, $S' \in \mathcal{M}(S)$ if and only if there exists a bijection $\sigma : S \rightarrow S'$ such that $\sigma(v) \in \Gamma^+(v), \forall v \in S$ where $\Gamma^+(v) = \{w \in V : (v, w) \in E\}$. A vertex subset $S \subseteq V$ is called *independent* if there exists an infinite sequence $(S_i)_{i \in \mathbb{N} \setminus 0}$ with $S_1 = S$ and $S_{i+1} \in \mathcal{M}(S_i), \forall i$. Notice that an independent set is not necessarily a stable set (we are here using the terminology related to matroid

theory). S is independent if it is possible to find $|S|$ infinitely long walks, one starting at each vertex of S , such that for each integer number k , the vertices appearing in the k^{th} position in the $|S|$ walks are all different. Assume $|S|$ vehicles moving on a (directed) network: they start from S (one vehicle at each vertex of S) and at every step they move simultaneously to a neighbour vertex, we want to study if they can do this infinitely many times without ever meeting at any vertex. Independent sets are subsets for which this is possible.

We note that the *meeting* problem on graphs has gained its interest previously. Various results have been obtained in the area of merging or meeting of two or more tokens (or vehicles / robots in the literature). In particular, the interested reader can refer to a (deterministic) rendezvous problem where two given robots should meet at some point, given a set of instructions (see for example, [1]). The meeting time for random walks and an arbitrary number of tokens (for connected undirected graphs) has been also evaluated [2]. The authors however, are not aware of any works devoted to the contrary, i.e., *no-meet* problem which we state and solve in this paper.

The set of all independent subsets in G is denoted by \mathcal{I} . The number of vertices of G is denoted by n .

Let $N(G)$ (N for no-meet) be the pair (V, \mathcal{I}) . The paper is dedicated to the study of $N(G)$. We will prove in Section 2 that $N(G)$ is a matroid. Then we show in Section 3 that the independence oracle is polynomial-time. A more efficient algorithm is proposed in Section 4 to compute the size of a basis of the matroid. Section 5 is dedicated to mixed graphs where we study the orientation problem of undirected edges to minimize or to maximize the rank of some set in the related matroid. Connections with other classes of matroids are investigated in Section 6. Extensions and applications are presented in Section 7. Further open questions are proposed in the last section.

2 $N(G)$ is a matroid

Unless otherwise stated, we assume that $G = (V, E)$ is a digraph on n vertices and $S \subseteq V$. We are going to prove that $N(G)$ is a matroid. It will then be called *the no-meet matroid*.

Let L denote a positive integer. Let B denote the digraph having for vertex set $V^1 \cup V^2$ with $V^i = \{v^i : v \in V\}$ for $i = 1, 2$ and for edge set $E = \{(v^1, v^2) : v \in V\}$. For $i = 1, 2, \dots, L$, $B^i = (V(B^i), E(B^i))$ denotes a copy of B . The vertices of B^i are denoted v_i^1 (resp. v_i^2) for the i -th copy of v^1 (resp. v^2). Then, let G_S denote the digraph with

- vertex set $V(G_S) = \{s, t\} \cup \bigcup_{i=1}^L V(B^i)$
- edge set $E(G_S) = \bigcup_{i=1}^L E(B^i) \cup \{(s, v_1^1) : v \in S\} \cup \{(v_L^2, t) : v \in V\} \cup \bigcup_{i=1}^{L-1} \{(v_i^2, w_{i+1}^1) : (v, w) \in E\}$.

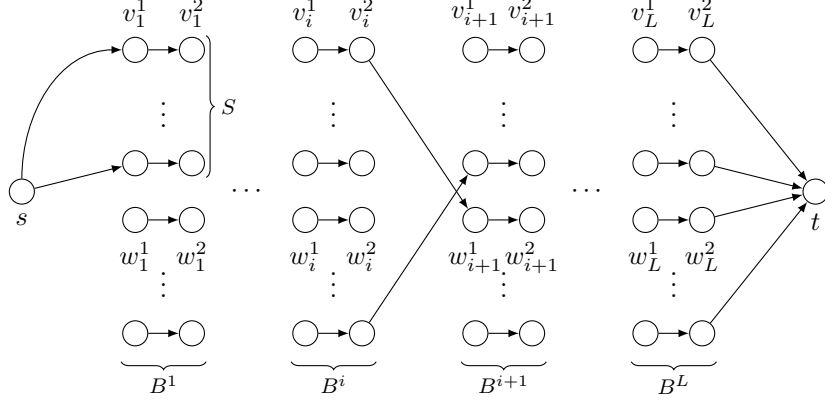


Figure 1: Construction of the graph G_S . For readability, only two edges of G are represented, including the edge (v, w) .

G_S will also be seen as a flow network where the capacity of each edge is set to 1.

Proposition 2.1. *Let $S \subseteq V$ and set $L := 1 + \binom{n}{|S|}$. The set S is independent if and only if the maximum st -flow value in G_S is $|S|$.*

Proof. [\Leftarrow]. Assume that there exists a maximum st -flow f in G_S of value $|S|$. For each $i \in \{1, 2, \dots, L\}$ define $W_i = \{v \in V : f(v_i^1, v_i^2) = 1\}$. We then have $W_1 = S$. Due to the number L of layers and to the capacity constraints, there must exist two indices i_1, i_2 such that $W_{i_1} = W_{i_2}$. This implies the existence of an infinite sequence $(S_i)_{i \in \mathbb{N}^*}$ with $S_1 = S$ and $S_{i+1} \in \mathcal{M}(S_i), \forall i$ (e.g. considering the repetition of the sequence $W_{i_1}, W_{i_1+1}, \dots, W_{i_2}$ after $S = W_1, \dots, W_{i_1-1}$).

[\Rightarrow]. Assume that S is an independent set. Then an st -flow of value $|S|$ in G may be defined as follows from any infinite sequence of independent sets $(S_i)_{i \in \mathbb{N}^*}$ starting with S . Consider then the first L sets of the sequence:

$$S = S_1 \rightarrow S_2 \rightarrow S_3 \rightarrow \dots \rightarrow S_L.$$

Let σ^i denote a bijection from S_i to S_{i+1} as described in the introduction, for $i = 1, \dots, L-1$. Then a maximum flow in G_S is obtained by sending a unit flow on each of the $|S|$ paths (represented by their set of vertices):

$$\left(s, v_1^1, v_1^2, (\sigma^1(v))_2^1, (\sigma^1(v))_2^2, \dots, (\sigma^{L-1}(\sigma^{L-2}(\dots(v))))_L^1, t \right), v \in S. \quad \square$$

Theorem 2.2. $N(G) = (V, \mathcal{I})$ is a matroid.

Proof. Following the proof of Proposition 2.1, one can easily check that if S is independent, then the same holds for any subset of S . So (V, \mathcal{I}) is an independence system. We now prove that the augmentation property holds for (V, \mathcal{I}) . Let $T \subset V$ such that $|S| < |T|$ and $T \in \mathcal{I}$.

To do so, we build an auxiliary graph H_S which is obtained from G_S by adding a vertex s' and the following set of edges, all having unit capacity: (s, s') and $(s', v_1^1), \forall v \in T \setminus S$. The number of layers in H_S is $L = 1 + \binom{n}{|S|}$.

Assume that the minimum cut value in H_S is less than or equal to $|S|$, and let \mathcal{C} denote such a minimum cut in H_S . If $(s, s') \in \mathcal{C}$, then this would imply the existence of an st -cut in G_S of size at most $|S| - 1$, a contradiction with S being independent (see Proposition 2.1). So (s, s') does not belong to any minimum cut in H_S . But then, the minimum cut value in H_S is also the same as the one of $G_{T \cup S}$ (since by contracting the nodes s and s' in H_S we get $G_{T \cup S}$). Since T is assumed to be independent, then by Proposition 2.1, the minimum cut value cannot be less than $|T| > |S|$, a contradiction.

So there exists in H_S an integral flow of value $|S| + 1$. And by Proposition 2.1, it follows that $S \cup \{z\}$ is independent with $z = \{w: f(s', w_1^1) = 1, w \in T \setminus S\}$, where f denotes a maximum flow in H_S . \square

To illustrate the independent sets of the matroid, let us consider the graph of Figure 2 (left part) where the outdegree of each vertex is equal to 1. We also show the graphs G_2 and G_3 obtained from G as follows. The vertices of G_2 correspond with subsets of vertices of G of size 2. If $S' \in \mathcal{M}(S)$ with $|S| = |S'| = 2$ then G_2 contains an edge from S to S' . If the outdegree of some vertex S (a subset S is identified with the vertex representing it in G_2) is equal to 0, then all edges going into S are eliminated since going through S will never help to identify independent sets. This procedure is recursively repeated to simplify G_2 . Observe for example that the set $\mathcal{M}(\{1, 6\}) = \emptyset$ and that is why there is no edge from $\{5, 3\}$ to $\{1, 6\}$ even if $\{1, 6\} \in \mathcal{M}(\{5, 3\})$. Independent sets of size 2 are then simply the sets that are not isolated in G_2 . Thus, $\{3, 5\}$, $\{2, 4\}$ and $\{1, 6\}$ are the only dependent sets of size 2. In fact, $\{3, 5\}$, $\{2, 4\}$ and $\{1, 6\}$ are the circuits of $N(G)$ (minimal dependent sets). G_3 is constructed in similar way by considering subsets of size 3. One can see here that there are 8 independent sets of size 3: $\{1, 2, 3\}$, $\{1, 2, 5\}$, $\{1, 3, 4\}$, $\{1, 4, 5\}$, $\{2, 3, 6\}$, $\{2, 5, 6\}$, $\{3, 4, 6\}$ and $\{4, 5, 6\}$. In fact, each one of these 8 subsets is a basis (a maximal independent set). This means that if we build G_4 , we get a graph with 0 edges (after application of the recursive edge elimination procedure described above). Observe that there is a loop involving $\{1, 2, 3\}$ since $\{1, 2, 3\} \in \mathcal{M}(\{1, 2, 3\})$. In fact, it is easy to show that for any digraph G for which the outdegree of each vertex is equal to 1, each connected component contains exactly one directed cycle whose size is equal to the size of maximal independent sets inside the connected component. The basis size is just the sum of cycle sizes through the connected components of the graph. It is also a simple exercise to show that all circuits are of size 2. A subset $S = \{v, w\}$ of size 2 is a circuit if the paths starting from v and w (these paths are unique here since the outdegree is 1) meet inside the directed cycle. In the example of Figure 2 there is only one connected component and the directed cycle goes through vertices 1, 2 and 3. The sets $\{3, 5\}$, $\{2, 4\}$ and $\{1, 6\}$ are obviously circuits. More generally, to check whether a subset $S = \{v, w\}$ is a circuit where v and w belong to the same connected component, one can select any vertex u inside

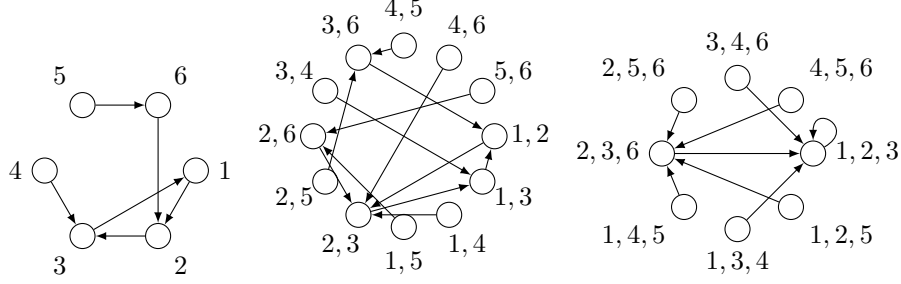


Figure 2: A digraph G where the outdegrees are equal to 1, and the corresponding digraphs G_2 and G_3 . For readability, nodes with degree 0 in those two graphs were removed.

the cycle (for example, vertex 1 in our case) and compute $l(v, u)$ and $l(w, u)$ respectively denoting the length of the unique path from v to u and from w to u . Then $S = \{v, w\}$ is a circuit if and only if v and w belong to the same connected component and $l(v, u) \equiv l(w, u) \pmod{l_c}$ where l_c is the length of the directed cycle contained in the same component as v and w . In the example of Figure 2, we have $l_c = 3$, $l(5, 1) = 4$ and $l(3, 1) = 1$ implying that $\{3, 5\}$ is a circuit. The example of graphs where outdegrees are equal to 1 is then easy to handle: circuits are easy to find and their number is quadratically bounded, the size of the basis is just the sum of size of directed cycles, and a set is independent if and only if it does not contain any circuit.

3 Independence oracle

We have seen at the end of Section 2 that one can easily determine whether a set is independent when outdegrees are equal to 1. In the general case, we can either use the characterization provided in Proposition 2.1 or try to build graphs of type G_k as described in Section 2. However, it might seem that we should assume that $|S|$ is upper bounded by a constant not depending on the instance to get polynomial-time algorithms (otherwise both the number of layers in the graph of Figure 1 and the number of vertices in G_k with $k = |S|$ are not polynomially bounded). We will nevertheless prove that there is a polynomial-time independence oracle by showing that the number of layers considered in the graph G_S can be polynomially bounded. We will first prove the result for strongly-connected digraphs and then generalize it to all digraphs.

Let us introduce some notation. Consider the standard Boolean operators \oplus (the "or" operator) and \otimes (the "and" operator). Given two Boolean matrices A and B where the number of columns of A and the number of rows of B are both equal to some p , $A \otimes B$ denotes the matrix where $(A \otimes B)_{ij} = \bigoplus_{k=1}^p a_{ik} \otimes b_{kj}$. The k^{th} Boolean power of a square matrix A is then $A^{\otimes k} = A^{\otimes k-1} \otimes A = A \otimes A^{\otimes k-1}$.

If A represents the adjacency matrix of a digraph G , then $(A^{\otimes k})_{ij} = 1$ if and only if there is in G a walk of length k from i to j . Since the number of Boolean n -square matrices is finite, some cyclicity should be expected when we compute the powers of the adjacency matrix. This can be precisely stated by considering the index of imprimitivity (sometimes called the index of cyclicity) defined as the greatest common divisor of the lengths of all the directed cycles of the given graph [3]. The index of imprimitivity is obviously less than or equal to n where n is the number of vertices.

Lemma 3.1. [3] *Let $G = (V, E)$ be a strongly connected digraph. Let A be the adjacency matrix of G , $c = c(A) > 0$ the index of imprimitivity and n the number of vertices of G , then there exists some number $k_{n,c}$ defined by*

$$k_{n,c} = \begin{cases} (n-1)^2 + 1 & \text{if } c = 1 \\ \max(n-1, \frac{n^2-1}{2} + \frac{n^2}{c} - 3n + 2c) & \text{if } c \geq 2, \end{cases} \quad (1)$$

such that $A^{\otimes k+c} = A^{\otimes k}$ for all $k \geq k_{n,c}$.

Roughly speaking, since $c \leq n$, we can say from Lemma 3.1 that for k above $O(n^2)$, there exists $k' < k$ such that $A^{\otimes k'} = A^{\otimes k}$.

Proposition 3.2. *Assume that G is a strongly-connected digraph. Independence can be checked in polynomial time.*

Proof. Let S be a subset of vertices and consider the graph G_S introduced in Section 2 and represented in Figure 1. Consider a minimum cut separating s and t . The size of the cut is less than or equal to $|S|$. Suppose that (v_i^2, w_{i+1}^1) belongs to such cut. Since all s - t paths going through (v_i^2, w_{i+1}^1) necessarily contain (v_i^1, v_i^2) , we can replace (v_i^2, w_{i+1}^1) by (v_i^1, v_i^2) . An edge of type (s, v_1^1) can also be replaced by the edge (v_1^1, v_1^2) . In other words, we can assume that the minimum cut contains only edges inside the graphs B^i . The edges of the minimum cut can be numbered e_1, e_2, \dots, e_l (with $l \leq |S|$). For $j = 1, \dots, l$, let i_j be the index such that $e_j \in B^{i_j}$.

Among all possible minimum cuts separating s and t , consider one minimizing the largest index of graphs B^i containing one edge of the cut. Said another way, B^{i_l} contains an edge of the minimum cut (e_l) while B^i does not contain any such edge for $i > i_l$, and we consider a minimum cut for which the index i_l is as small as possible.

We may assume that $i_1 \leq i_2 \leq \dots \leq i_l$. Suppose that for some j we have $e_j \in B^{i_j}$, $e_{j+1} \in B^{i_{j+1}}$ and $i_{j+1} > i_j$. Observe that the walks between vertices of B^{i_j} and vertices of $B^{i_{j+1}}$ can be seen as walks of length $i_{j+1} - i_j$ in the original graph G . From Lemma 3.1 (and the remark following it), since we are minimizing i_l , the gap $i_{j+1} - i_j$ cannot be above $O(n^2)$ due to the cyclic behaviour of the powers of the adjacency matrix A . We give here a formal proof of this fact.

Assume for a contradiction that $i_{j+1} - i_j = k + c$, for some integer k such that $k \geq k_{n,c}$, with c and $k_{n,c}$ as described in Lemma 3.1. Given any edge

$e = (v_i^1, v_i^2)$ in B^i with $i \geq i_{j+1}$, let \overleftarrow{e} denote the edge (v_{i-c}^1, v_{i-c}^2) in B^{i-c} . Similarly, given any edge $e = (v_i^1, v_i^2)$ in B^i with $i \leq L - c$, let \overrightarrow{e} denote the edge (v_{i+c}^1, v_{i+c}^2) in B^{i+c} .

Claim. The set of edges $\overleftarrow{\mathcal{E}} = \{e_1, e_2, \dots, e_j, \overleftarrow{e}_{j+1}, \overleftarrow{e}_{j+2}, \dots, \overleftarrow{e}_l\}$ defines an st -cut in G_S .

Proof of the claim.

Assume for contradiction that $\overleftarrow{\mathcal{E}}$ does not define an st -cut in G_S . Then there exists some st -path \mathcal{P} in G_S not containing any edge in $\overleftarrow{\mathcal{E}}$. Let $(p_{i_l-c}^1, p_{i_l-c}^2)$ denote the edge of \mathcal{P} contained in B^{i_l-c} . Also, let $(w_{i_j}^1, w_{i_j}^2)$ (resp. $(z_{i_j+k}^1, z_{i_j+k}^2)$) denote the edge of \mathcal{P} in B^{i_j} (resp. B^{i_j+k}). Assume that $\overleftarrow{e}_{j+1} = (r_{i_j+k}^1, r_{i_j+k}^2)$. Then, since $\overleftarrow{e}_{j+1} \notin \mathcal{P}$, necessarily $z \neq r$. Let \mathcal{P}_1 (resp. \mathcal{P}_2) denote the part of the path \mathcal{P} from s to $w_{i_j}^2$ (resp. from $z_{i_j+k}^1$ to $p_{i_l-c}^2$). Since $A^{k+c} = A^k$, there also exists a path Q in G_S from $w_{i_j}^2$ to $z_{i_{j+1}}^1 = z_{i_j+k+c}^1$ not containing any edge in \mathcal{E} . Assuming that \mathcal{P}_2 corresponds to the sequence of edges $(f_i)_{i \in I}$ for some index set I , let $\overrightarrow{\mathcal{P}}_2$ denote the path corresponding to the sequence of edges $(\overrightarrow{f}_i)_{i \in I}$. Since \mathcal{P}_2 does not intersect $\overleftarrow{\mathcal{E}}$, $\overrightarrow{\mathcal{P}}_2$ does not intersect \mathcal{E} . Then, the concatenation of \mathcal{P}_1 , Q and $\overrightarrow{\mathcal{P}}_2$ is a path from s to $p_{i_l}^2$ in G_S not containing any edge in \mathcal{E} . Since \mathcal{E} does not contain any edge in B^i for $i > i_l$ and since any node different from t has outdegree at least one in G_S there must exist a path from $p_{i_l}^2$ to t in G_S . The concatenation of this path together with \mathcal{P}_1 , Q and $\overrightarrow{\mathcal{P}}_2$ leads to an st -path in G_S not intersecting \mathcal{E} , a contradiction with \mathcal{E} being an st -cut in G_S . This terminates the proof of the claim.

It follows from the last claim that $i_{j+1} - i_j < k_{n,c} + c$, for all $j \in \{1, 2, \dots, L-1\}$. Also, with an argumentation similar to the one used in the proof of the claim we can easily show that necessarily $i_1 < k_{n,c} + c$.

In the worst case $i_{j+1} - i_j = O(n^2)$. Since this holds for any pair of consecutive edges of the cut and also for the first edge (if e_1 is in B_{i_1} , then $i_1 = O(n^2)$), we deduce that $i_l = O(|S| \cdot n^2) = O(n^3)$. The number of layers that we should consider in G_S is then only $O(n^3)$ implying the polynomiality of the independence oracle. \square

To handle the general case of graphs containing several strongly-connected components, one might try to use a generalization of Lemma 3.1 found in [3]. The index of imprimitivity of general graphs is defined as the least common multiple of the nonzero imprimitivity indexes of all strongly-connected components of the graphs. A theorem of [3] states that $A^{\otimes k+c} = A^{\otimes k}$ for all $k \geq 2n^2 - 3n + 2$. However, the best upper bounds for the index of imprimitivity c are only of exponential type for general graphs [4]. Another approach is then needed to generalize the previous proposition.

We can see from the proof of Proposition 3.2 that given a strongly-connected

digraph G , the number of layers does not need to be more than $O(n^3)$. This is in fact equivalent to say that starting from a dependent set S , any set of $|S|$ walks will necessarily meet in $O(n^3)$ steps.

Corollary 3.3. *Assume that G is strongly-connected and let S be a dependent set of $N(G)$. Any set of $|S|$ walks starting at S will necessarily meet after at most $O(n^3)$ steps.*

Proof. By Proposition 2.1, since S is assumed to be dependent, the maximum st -flow value in G_S is at most $|S|-1$. By the proof of Proposition 3.2, there exists a minimum cut in G_S that is composed of edges contained in the subgraphs B^i of G_S with $i = O(n^3)$. It follows that for any set of $|S|$ st -paths in G_S at least two of them must intersect after at most $O(n^3)$ layers. This is equivalent to say that at least two of the corresponding walks in G meet after at most $O(n^3)$ steps. \square

Another way to rephrase the corollary is to say that there exists a constant C such that for any strongly-connected digraph $G = (V, E)$, for any set of vertices $S \subseteq V$ that is dependent in $N(G)$, and for any set of $|S|$ walks starting at S , there are at least two walks that contain the same vertex at some position $k \leq Cn^3$ (i.e., they meet).

Assume that G has r strongly-connected components H_1, H_2, \dots, H_r . Let $S \subseteq V$ be a subset of vertices of G . Let $S^j \subseteq S$ be the set of vertices of S belonging to the strongly-connected component H_j .

Lemma 3.4. *If S is dependent in $N(G)$, then there exists at least one component H_j such that $S^j = S \cap V(H_j)$ is dependent in $N(H_j)$.*

Proof. If S^j is independent, then one can build $|S^j|$ walks inside H_j starting from S^j that never meet (in the sense introduced in Section 1). If all S^j are independent, then we can build no-meeting $|S|$ walks starting from S , contradicting the dependency of S . \square

Theorem 3.5. *Let G be any digraph. Independence can be checked in polynomial time.*

Proof. Let $S = \{v_1, v_2, \dots, v_{|S|}\}$ be any dependent set. Consider a longest sequence $S = S_1 \rightarrow S_2 \rightarrow S_3 \rightarrow \dots \rightarrow S_L$ starting from S such that $S_{i+1} \in \mathcal{M}(S_i)$. We obviously have $\mathcal{M}(S_L) = \emptyset$ and $S_i \neq S_j$ for any $i < j$. Also, to the sequence $(S_i)_{i=1}^L$ we can associate a set of $|S|$ walks $(P^q)_{q=1}^{|S|}$ with P^q corresponding to the sequence of vertices $(w_i^q)_{i=1}^L$ such that $w_1^q = v_q$ and w_i^q is the vertex appearing in position i for $i = 1, \dots, L$ in the walk P^q .

Assume that G has r strongly connected components H_1, \dots, H_r . Since the graph G' obtained by shrinking these strongly-connected components does not contain directed cycles, once a walk P^q leaves a component H_j , it can never go back to it. This implies that the number of indices i for which w_i^q and w_{i+1}^q do not belong to the same component is at most $r - 1$. Consequently, there are at most $(r - 1) \cdot |S| = O(n^2)$ indices i among $\{1, \dots, L\}$ for which there exists at

least one walk P^q moving from one component to another one.

Let us consider two indices a and b for which such moves occur and such that there are no moves between a and b . In other words, for each $q \in \{1, \dots, |S|\}$ and each index $i \in \{a+1, \dots, b-1\}$, w_i^q and w_{i-1}^q belong to the same component while there exists at least one index q' (resp. q'') such that $w_a^{q'}$ and $w_{a-1}^{q'}$ (resp. $w_b^{q''}$ and $w_{b-1}^{q''}$) are not in the same component.

Let $S_a^j = S_a \cap H_j$. From Lemma 3.4, we know that there is at least one index $\bar{j} \in \{1, 2, \dots, r\}$ such that $S_a^{\bar{j}}$ is dependent in $N(H_{\bar{j}})$. Then, from Corollary 3.3, we necessarily have $b - a = O(n^3)$.

We consequently have $L = O(n^2 \cdot n^3) = O(n^5)$. This means that the number of layers in the graph G_S does not need to be more than $O(n^5)$. \square

4 Basis size

The size of a basis can obviously be computed using the independence oracle described in Section 3. A more efficient algorithm is described in this section.

Following [5], a cycle subdigraph is defined as a collection of vertex-disjoint directed cycles $\{C_1, \dots, C_k\}$ of $G = (V, E)$, such that $V(C_i) \subseteq V$ and $V(C_i) \cap V(C_j) = \emptyset$ for every $i, j = 1, \dots, k$. Notice that a cycle subdigraph does not need to span all graph vertices. The size of the cycle subdigraph is the number of edges contained in the cycles of the collection.

Let $d^+(v)$ (resp. $d^-(v)$) denote the number of edges leaving (resp. entering) v . Let us first recall this well-known decomposition result of k -regular digraphs (a digraph where $d^+(v) = d^-(v) = k$ for any $v \in V$). Notice that loops and parallel edges are allowed. The proof is given for sake of completeness.

Lemma 4.1. *The set of edges of every k -regular digraph $G = (V, E)$ can be partitioned into the edge sets of k cycle subdigraphs of G .*

Proof. Build a bipartite graph B by replacing each vertex $v \in V$ by two vertices v' and v'' and adding edge (v', w'') whenever $(v, w) \in E$. This bipartite graph is k -regular. From König-Hall Theorem, B contains a perfect matching. Then by deleting the edges of the perfect matching we get another regular bipartite graph for which we can apply the same decomposition result. This leads to a decomposition of the set of edges of B into k edge-disjoint perfect matchings. Each perfect matching of B is obviously equivalent to a cycle subdigraph of G . \square

Lemma 4.2. *Let $G = (V, E)$ be a digraph such that $d^+(v) = d^-(v) \leq \delta$ for any $v \in V$. Then there is a cycle subdigraph of G containing at least $\left\lceil \frac{|E|}{\delta} \right\rceil$ edges of G .*

Proof. It is possible to build a δ -regular digraph G' from G , by simply adding enough loops. From Lemma 4.1, the edges of G' can be partitioned into the edge sets of δ cycle subdigraphs (of G'). Among these cycle subdigraphs, there must

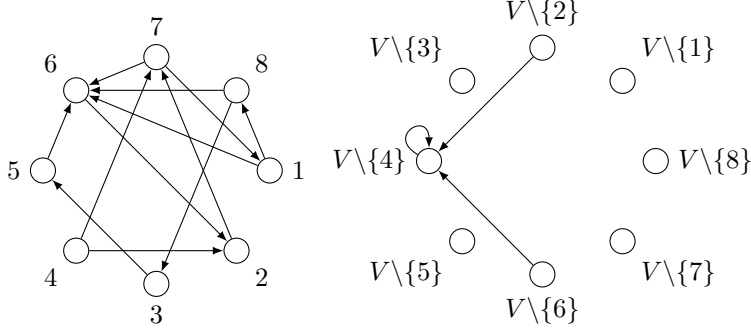


Figure 3: A digraph $G = (V, A)$, and the corresponding digraph G_7

be one using at least $\left\lceil \frac{|E|}{\delta} \right\rceil$ edges of G . Notice this cycle subdigraph might contain some of the loops that have been added. However, by deleting the loops we still get a cycle subdigraph of G and the number of edges inside G does not change since the deleted edges were not in G . \square

Theorem 4.3. *Let $G = (V, E)$ be a digraph. The size of a basis of $N(G)$ (i.e., $\text{rank}(V)$) is equal to the size of the largest cycle subdigraph of G .*

Proof. It is straightforward to verify that the set of vertices of a cycle subdigraph X is an independent set since $V(X) \in \mathcal{M}(V(X))$.

Let S be an independent set. Let $S_1 = S, S_2, \dots, S_i, \dots$ be an infinite sequence of independent sets such that $S_{i+1} \in \mathcal{M}(S_i)$. Since the number of subsets of size $|S|$ is limited, this sequence contains at least one cyclic subsequence, say $S_{i_0}, S_{i_0+1}, \dots, S_{i_0+c-1}, S_{i_0+c} = S_{i_0}$.

Let σ^i be the bijection from S_i to S_{i+1} . Remember that for each $v \in S_i$, $\sigma^i(v) \in S_{i+1}$ and $(v, \sigma^i(v)) \in E$.

Consider the digraph F having as vertex set $S_{i_0} \cup \dots \cup S_{i_0+c-1}$ and as edge set all the edges of G related to bijections σ^i (for $i_0 \leq i \leq i_0 + c - 1$). An edge that appears in μ bijections has μ parallel copies in F . Observe that F contains $|S| \cdot c$ edges. Moreover, for each vertex $v \in V(F)$, we have $d^+(v) = d^-(v)$ and both are equal to the number of subsets $S_{i_0}, \dots, S_{i_0+c-1}$ containing v . This immediately implies that $d^+(v) = d^-(v) \leq c$. Therefore by Lemma 4.2, F contains a cycle subdigraph X' such that $|V(X')| \geq \frac{|S| \cdot c}{c}$. Notice that a cycle subdigraph of F is also a cycle subdigraph of G (even if F can contain different copies of the same edge, a cycle subdigraph cannot contain more than a copy of any edge). \square

Computing a largest cycle subdigraph is a well-know problem that can be solved, for example, by computing a maximum weight perfect matching in a bipartite graph whose complexity is $O(n^3)$ [5]

To illustrate the results above, we consider a directed graph having 8 vertices and 12 edges (Figure 3). The set of bases of $N(G)$ contains $\{1, 2, 3, 4, 5, 7, 8\}$,

$\{1, 2, 3, 5, 6, 7, 8\}$ and $\{1, 3, 4, 5, 6, 7, 8\}$. We also show the graph G_7 (obtained as described at Section 2). Observe that there is a loop around the set $\{1, 2, 3, 5, 6, 7, 8\}$ which corresponds to the vertex set of a cycle subdigraph. $N(G)$ has here only one circuit given by the set $\{2, 4, 6\}$.

5 Mixed-graph orientation for no-meet

Given an undirected graph H , one might be interested in orienting the graph to get a directed graph \bar{H} such that the size of the basis of $N(\bar{H})$ is maximized. This problem is easy to solve since it can be seen as a maximum-weight 2-factor problem (by adding 0-weight loops to H) and transformed into a maximum matching using the construction of Tutte [6]. Then we only have to orient the edges of the undirected cycles to get a cycle subdigraph.

A less trivial problem is to orient the edges of some mixed graph such that some given subset S is independent in the corresponding no-meet matroid. More formally, let $H = (V, \vec{E}, U)$ be a mixed graph where \vec{E} is a set of directed edges while U is a set of undirected edges. Given a subset of vertices $S \subseteq V$, we would like to check whether it is possible to orient the undirected edges U such that S is independent in $N(\bar{H})$ where \bar{H} is the obtained directed graph $\bar{H} = (V, \vec{E} \cup \vec{U})$. Let us call this problem, *the no-meet mixed graph orientation problem*.

Theorem 5.1. *Deciding whether there is an orientation of the undirected edges of a mixed graph such that a given subset S is independent in the related no-meet matroid is NP-complete.*

Proof. The decision problem is clearly in NP: given any orientation, it is possible to check independence in polynomial time using the algorithm of Section 3.

A reduction from 3-SAT is proposed here. Let us consider a 3-SAT formula that is a conjunction of m clauses C_1, \dots, C_m where each C_i is a disjunction of at most 3 variables among r Boolean variables x_1, \dots, x_r . A mixed graph H is built as follows. For each variable x_i , 4 vertices u_i, v_i, w_i and z_i are considered. An undirected edge $u_i v_i$ has to be oriented: if it is oriented from u_i to v_i , then x_i receives the value true, otherwise x_i is false. As shown in Figure 4, H contains the directed edges (u_i, w_i) and (v_i, z_i) . We add for each clause C_j two vertices s_j and t_j . Assume that clause C_j contains the variable x_i in positive form ($C_j = x_i \vee \dots$) while C_k contains x_i in negative form ($C_k = \bar{x}_i \vee \dots$). Then s_j is connected to u_i through a path of length a_j where a_j is a number that will be determined later. Two vertices $f_{j,i}$ and $g_{j,i}$ are also considered. $f_{j,i}$ is connected to w_i through a path of length $1 + a_j$ and a directed edge $(w_i, g_{j,i})$ is added. The graph H contains also an edge directed from z_i to t_j . Loops around t_j and $g_{j,i}$ are also considered. Since clause C_k contains \bar{x}_i , the vertices and edges relating C_k and x_i are symmetric to those defined for C_j : there is a path of length a_k from s_k to v_i , a path of length $1 + a_k$ connecting $f_{k,i}$ to z_i , an edge $(z_i, g_{k,i})$ and an edge (w_i, t_k) . The numbers a_j are chosen such that $a_j \neq a_k$ and $a_j \neq a_k + 1$ for any pair of clauses C_j and C_k . We can, for example, take $a_j = j^2$. Observe

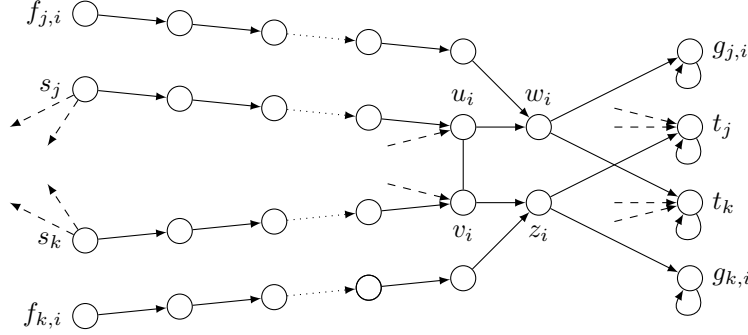


Figure 4: Gadget of the reduction from 3-SAT to the no-meet mixed graph orientation problem. Note that the nodes s_j , t_j , s_k and t_k are attached to two other such gadgets corresponding to their two other respective literals. Similarly, the nodes u_i and v_i are connected to other nodes s_l for each clause containing respectively x_i and \bar{x}_i .

that the number of vertices of H is at most $4r + 2m + 3 \sum_{j=1}^m (2a_j + 1) = O(m^3)$ implying the polynomiality of the construction.

Let $S = \bigcup_{j=1}^m \{s_j\} \cup \bigcup_{j=1, \dots, m}^{i=1, \dots, r} \{f_{j,i} : x_i \in C_j\} \cup \bigcup_{j=1, \dots, m}^{i=1, \dots, r} \{f_{j,i} : \bar{x}_i \in C_j\}$. Let us show that there is an orientation of the undirected edges $u_i v_i$ of the mixed graph H such that S is independent, if and only if, the 3-SAT formula can be satisfied.

Let us assume that the 3-SAT formula is satisfiable and consider a truth assignment satisfying it. As mentioned above, if the value of x_i is true, then edge $u_i v_i$ is oriented from u_i to v_i , otherwise it is oriented in the opposite direction. Let us build $|S|$ walks starting from S . First of all, for every j and every i such that $x_i \in C_j$ (resp. $\bar{x}_i \in C_j$) we can construct walks from $f_{j,i}$ going through w_i (resp. z_i), ending at $g_{j,i}$ and staying there forever (there is a loop in $g_{j,i}$). Since the 3-SAT formula is satisfied, there must exist for each clause C_j a literal such that either $x_i \in C_j$ and x_i is true or $\bar{x}_i \in C_j$ and x_i is false. In the first case, we consider the walk starting at s_j , going through u_i , v_i , z_i , t_j and staying forever in t_j (there is a loop in t_j). In the second case, when x_i appears in negative form in a C_j and x_i is false, then one can build a walk starting at s_j , going through v_i , u_i , w_i , t_j and staying there forever. These two families of walks do not meet (in the sense defined in this paper) the walks starting in $f_{j,i}$. Observe that when $j \neq k$ the two walks starting from s_j and s_k do not meet since $a_j \neq a_k$. Observe also that if the walk starting at s_j contains (u_i, v_i) , then it goes through z_i , but the walk starting from $f_{k,i}$ goes also through z_i (if C_k contains \bar{x}_i). However, since $a_j \neq 1 + a_k$, these walks do not reach z_i in the same time. Similarly, if the walk starting at s_j contains (v_i, u_i) , it will not meet the walk starting from $f_{k,i}$ (in w_i). The $|S|$ walks built in this way do not meet showing that S is independent in the matroid related to the directed graph obtained by orienting the edge $u_i v_i$ according to the truth values.

Let us now assume that there is an orientation such that S is independent in the related no-meet matroid. There is a walk starting from each vertex of S and these $|S|$ walks do not meet. Observe that, by construction, if x_i appears in positive (resp. negative) form in C_j , then the walk starting at s_j cannot go through edge (u_i, w_i) (resp. (v_i, z_i)), otherwise it will meet at w_i (resp. z_i) with the walk starting at $f_{j,i}$. Therefore, either the walk starting from s_j uses an edge (u_i, v_i) and $x_i \in C_j$, or it uses an edge (v_i, u_i) and $\bar{x}_i \in C_j$. It follows that the orientation defines a satisfying truth assignment. \square

A natural extension of the independence problem considered in Theorem 5.1 is to find an edge orientation maximizing the rank of a given subset $S \subseteq V$. Remember that the rank of set is the size of the largest independent subset contained in the set. Let us use $r_H^{max}(S)$ to denote such maximum rank (there exists an orientation such that $r_H^{max}(S) = rank(S)$ in $N(\bar{H})$ where \bar{H} is the directed graph obtained after the orientation of H).

To study the complexity of computing $r_H^{max}(S)$, we are going to use the PCP theorem [7]. Given a 3-SAT formula ϕ , let $val(\phi)$ be the maximum number of clauses that can be satisfied. The PCP theorem states that there is a constant $0 < \rho < 1$ such that for any language $L \in NP$, there is a mapping f from NP to 3-SAT instances, computable in polynomial time, such that $y \in L \implies val(f(y)) = m$ while $y \notin L \implies val(f(y)) < \rho m$ where m is the number of clauses of $f(y)$. Roughly speaking, it is NP-hard to distinguish between 3-SAT instances that can be satisfied and those for which it is not possible to satisfy a proportion ρ of its clauses.

Theorem 5.2. *There is a constant $0 < \alpha < 1$ such that it is NP-hard to approximate $r_H^{max}(S)$ within a ratio α .*

Proof. Let us consider again the reduction used in the proof of Theorem 5.1. We consider a 3-SAT formula ϕ , build the corresponding mixed graph, and take $S = \bigcup_{j=1}^m \{s_j\} \cup \bigcup_{j=1, \dots, m}^{i=1, \dots, r} \{f_{j,i} : x_i \in C_j\} \cup \bigcup_{j=1, \dots, m}^{i=1, \dots, r} \{f_{j,i} : \bar{x}_i \in C_j\}$.

Since each clause contains either 1, 2 or 3 literals, $|S| = m + m'$ where $m \leq m' \leq 3m$.

Is is also easy to check from the proof of Theorem 5.1 that given any orientation and the corresponding truth assignment, $rank(S)$ is equal to m' plus the number of clauses that are satisfied. Indeed, if $S' \subseteq S$ is a maximum independent set (so $|S'| = rank(S)$), then for each clause C_j that is not satisfied, we can assume that $s_j \notin S'$ (since otherwise, there is some $f_{j,i}$ that is not in S' , and S' can be modified by inserting $f_{j,i}$ and deleting s_j from it, leading to another independent set having the same size). In other words, S' contains all $f_{i,j}$ vertices in addition to s_j vertices corresponding to the satisfied clauses. As a consequence, we can write that $r_H^{max}(S) = val(\phi) + m'$.

Let us now assume that we have a polynomial-time α -approximation algorithm for the rank-maximization problem, where α is some positive constant $0 < \alpha < 1$. Let $\tilde{r}(S)$ be the value provided by the approximation algorithm. Then we have $r_H^{max}(S) \geq \tilde{r}(S) \geq \alpha r_H^{max}(S)$.

If $\tilde{r}(S) \geq m' + m\rho$, then $r_H^{max}(S) \geq m' + m\rho$. Using $r_H^{max}(S) = val(\phi) + m'$, we get that $val(\phi) \geq m\rho$.

On the other hand, if $\tilde{r}(S) < m' + m\rho$, we can deduce that $\alpha r_H^{max}(S) < m' + m\rho$ leading to $r_H^{max}(S) < \frac{m' + m\rho}{\alpha}$. Let us take $\alpha = \frac{3+\rho}{4}$. Observe that $\alpha \geq \frac{m' + m\rho}{m' + m}$ for any value of m' such that $m \leq m' \leq 3m$. This immediately implies that $r_H^{max}(S) < m' + m$ which is equivalent to $val(\phi) < m$.

Consequently, an α -approximation algorithm (with $\alpha = \frac{3+\rho}{4}$) allows to distinguish between 3-SAT instances that can be fully satisfied and those for which the proportion of satisfied clauses is less than ρ . We know from PCP theorem that this is NP-hard, showing the difficulty of the approximation within α . \square

The NP-hardness results holds for general subsets S . In fact, we can see in the reduction above that the sets S considered there are of arbitrary large size. One can wonder whether computing $r_H^{max}(S)$ is still NP-hard even if the size of S is bounded by some constant. We answer this question when $|S| = 1$.

Proposition 5.3. $r_H^{max}(S)$ can be computed in polynomial time when $|S| = 1$.

Proof. Let $H = (V, \vec{E}, U)$ be the mixed graph and let $S = \{s\} \subset V$. Let G be the digraph $G = (V, \vec{E} \cup U')$ where each undirected edge $e = uv \in U$ is replaced in U' by two directed edges (u, v) and (v, u) . Let W be the set of vertices that can be reached from s in G . For each vertex $u \in W$, we check in polynomial time whether u belongs to a directed cycle of G . One can proceed as follows: for each directed edge $(u, v) \in \vec{E} \cup U'$, we delete (u, v) and we check the existence of a path from v to u in G . Notice that if $(u, v) \in U' \setminus \vec{E}$ and $(v, u) \notin \vec{E}$, we also delete (v, u) from G when we are looking for a path from v to u (this is to avoid cycles containing only the two edges (u, v) and (v, u) ; observe however that such a cycle is accepted if either (u, v) or (v, u) belongs to \vec{E}).

Assume that there exists at least one vertex $u \in W$ belonging to a directed cycle C of G as defined above. Since $u \in W$, there exists a path P in G joining s to u . Let u' be the first common vertex of C and P and let P' be the subpath of P connecting s to u' . Since P' and C do not have common edges, each edge $ab \in U$ such that $(a, b) \in P' \cup C$ can be oriented from a to b . This means that there exists an orientation allowing the existence of an infinitely long walk starting at s . In other words, $r_H^{max}(S) = 1$.

It is also obvious that if all vertices of W do not belong to any directed cycle in G , then $S = \{s\}$ cannot be independent, implying that $r_H^{max}(S) = 0$. \square

Another natural extension is to find an edge orientation *minimizing* the rank of a given subset $S \subseteq V$. Let $r_H^{min}(S)$ denote such minimum rank.

Theorem 5.4. Computing $r_H^{min}(S)$ is NP-hard.

Proof. We use again the reduction of Theorem 5.1. As observed in the proof of Theorem 5.2, given any orientation and the corresponding truth assignment, $rank(S)$ is equal to m' plus the number of clauses that are satisfied. It is then clear that minimizing $rank(S)$ is equivalent to minimizing the number

of satisfied clauses. The NP-hardness of Min 3-SAT [8] implies the wanted result. \square

One can similarly study the complexity of computing $r_H^{min}(S)$ when $|S| = 1$.

Proposition 5.5. *$r_H^{min}(S)$ can be computed in polynomial time when $|S| = 1$.*

Proof. Let $H = (V, \vec{E}, U)$ be the mixed graph and let $S = \{s\} \subset V$. Let $W \subseteq V$ be the set of vertices that can be reached from s in the digraph $G = (V, \vec{E})$ (i.e., there is at least one directed path from s to each vertex of W). Let G^W be the subgraph of G induced by W : $G^W = (W, \vec{E}_W)$ where $\vec{E}_W \subseteq \vec{E}$ is the set of directed edges whose two extremities are in W . If G^W contains a directed cycle, then there is already a walk starting at s and reaching a directed cycle. This clearly implies that the rank of $\{s\}$ is equal to 1, and this rank does not depend on the orientation of the undirected edges. Consequently, $r_H^{min}(S) = 1$.

Let us now assume that G^W does not contain any directed cycle. Consider then a topological ordering of the vertices of G^W : w_1, \dots, w_l . Notice that $w_1 = s$ since every vertex of W can be reached from s and G^W is acyclic. Let us orient the edges of U as follows. If $e = w_i w_j$ with $j > i$, then e is oriented from w_i to w_j . Moreover, if $e = uv$ with $u \in W$ and $v \in V \setminus W$, then orient e from v to u . Finally, if both extremities of an undirected edge e are in $V \setminus W$, then orient e in arbitrary way. One can easily see that in the obtained oriented graph \bar{H} , there is no any path starting from s that can reach any directed cycle. This clearly implies that there is an orientation such that $S = \{s\}$ is dependent, implying that $r_H^{min}(S) = 0$. \square

6 Connections with other matroids

We provide here some connections with some well-known matroids. Let us start with transversal matroids. A transversal matroid whose ground set is V can be defined through a bipartite graph $(V \cup U, E)$. A subset $I \subseteq V$ is independent if it can be matched with a subset of U . Our first result states that the class of transversal matroids is included in the class of no-meet matroids.

Theorem 6.1. *Each transversal matroid is a no-meet matroid.*

Proof. Consider a transversal matroid M defined as recalled above. Consider a basis B of M where $B = \{v_1, v_2, \dots, v_r\} \subseteq V = \{v_1, v_2, \dots, v_n\}$ and r is the rank of M . A well-known result of [9] states that we can assume that $|U| = r$ (see, e.g., Theorem 22.5 of [10]). Assume then that v_i is matched with u_i for $1 \leq i \leq r$.

Let us build a digraph $G = (V, E)$ as follows. For each vertex $v \in V$, E contains an edge (v, v_i) , if and only if, v and u_i are adjacent in the bipartite graph related to M . Observe that E contains a loop (v_i, v_i) for each $v_i \in B$ (i.e., for $1 \leq i \leq r$). We are going to show that $M = N(G)$. Observe that the rank of $N(G)$ is also r . Consider any basis $\{v_{\sigma_1}, \dots, v_{\sigma_r}\}$ of M and assume that v_{σ_i}

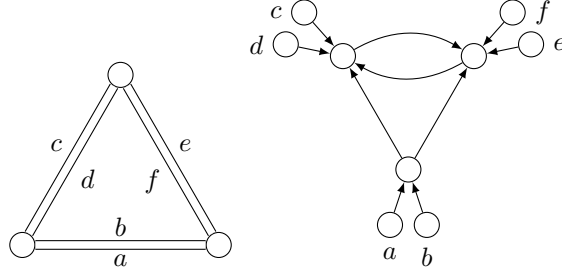


Figure 5: A laminar matroid that is not a no-meet (left); a no-meet matroid that is not transversal (right)

is matched with u_i for $1 \leq i \leq r$. Then G contains an edge from v_{σ_i} to v_i , and since there is a loop around each v_i for $1 \leq i \leq r$, we get that $\{v_{\sigma_1}, \dots, v_{\sigma_r}\}$ is a basis of $N(G)$. Conversely, assume that $\{v_{\sigma_1}, \dots, v_{\sigma_r}\}$ is a basis of $N(G)$. Then $\{v_1, \dots, v_r\}$ is a successor of $\{v_{\sigma_1}, \dots, v_{\sigma_r}\}$ (as defined in Section 1). We can then assume that E contains an edge (v_{σ_i}, v_i) for $1 \leq i \leq r$. This implies that each vertex v_{σ_i} is adjacent to u_i in the bipartite graph. The set $\{v_{\sigma_1}, \dots, v_{\sigma_r}\}$ is consequently a basis of the transversal matroid M . The matroids M and $N(G)$ have then the same ground set and the same set of bases. They are hence equal. \square

We prove now that the inclusion in Theorem 6.1 is strict.

Proposition 6.2. *There is a no-meet matroid that is not transversal.*

Proof. Consider the no-meet matroid whose underlying digraph is represented on the right part of Figure 5. Assume for contradiction that this matroid is transversal. Then its restriction to $\{a, b, c, d, e, f\}$ would be transversal (since the class of transversal matroids is restriction-closed). Observe that the matroid obtained through this restriction is exactly the graphic matroid shown on the left part of Figure 5. This matroid is however known to be non-transversal (see, e.g., Example 1.6.3 of [11]). \square

A laminar matroid (see, e.g., [12]) is defined through a laminar family \mathcal{F} , i.e., a collection of subsets of a set E such that, for any two intersecting sets, one is contained in the other. For a capacity function c on \mathcal{F} , a set $I \subset E$ is independent if $|I \cap F| \leq c(F)$ for all $F \in \mathcal{F}$.

Next proposition states that the set of no-meet matroids is not included in the set of laminar matroids.

Proposition 6.3. *There is a no-meet matroid that is not laminar.*

Proof. Consider the no-meet matroid related to the digraph represented on the right part of Figure 6. It is in fact equal to the graphic matroid represented on the left part of Figure 6. Observe that $\{a, d, e\}$ and $\{a, b, c\}$ are two circuits

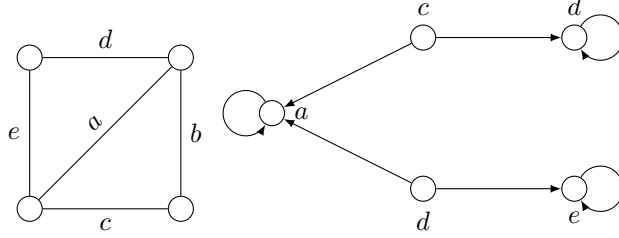


Figure 6: A no-meet matroid that is not laminar

of this matroid having a non-empty intersection. By a theorem of [12], two intersecting circuits C_1 and C_2 of a laminar matroid necessarily satisfy the following: the closure of C_1 is either contained in or containing the closure of C_2 . Observe that the two circuits $\{a, d, e\}$ and $\{a, b, c\}$ are closed (flat). They clearly do not satisfy the condition above, implying that the matroid is not laminar. \square

We also show that the set of laminar matroids is not a subset of no-meet matroids. We will use $\delta^+(a)$ to denote the set edges going out of a vertex a .

Proposition 6.4. *There is a laminar matroid that is not a no-meet matroid.*

Proof. Let us consider the graph of Figure 5 (left part). The graphic matroid related to this graph is laminar. Any independent set I should satisfy $|I \cap \{a, b\}| \leq 1$, $|I \cap \{c, d\}| \leq 1$, $|I \cap \{e, f\}| \leq 1$ and $|I \cap \{a, b, c, d, e, f\}| \leq 2$. Suppose for contradiction that this laminar matroid is a no-meet matroid $N(G)$ where $G = (V, E)$ is a digraph with $V = \{a, b, c, d, e, f\}$. Since the rank of the matroid is equal to 2, G contains a cycle subdigraph of size 2 (Section 4). Moreover, $N(G)$ does not contain any circuit of size 1 (a loop in the matroid sense) implying that the outdegree of each vertex in G is at least 1. Let us consider all possible alternatives.

Case 1. Assume that G does not contain loops. Then G contains one cycle of size 2. For symmetry reasons, we can assume that E contains (a, c) and (c, a) . Observe that if there is another edge in $\delta^+(a)$ (say (a, x) for some $x \notin \{a, c\}$), and since the outdegree of all vertices is at least 1, then G contains also the edge (x, a) (otherwise, we get a cycle subdigraph of size strictly larger than 2). The same conclusion holds for c . Observe however that if G simultaneously contains (a, x) , (x, a) , (c, y) and (y, c) for some $y \notin \{a, c\}$, then G contains two disjoint cycles of size 2 (or a cycle of size 3, when $x = y$) contradicting the rank condition. We can then assume that the outdegree of c is exactly 1 (i.e., the only edge of $\delta^+(c)$ is (c, a)). This implies that any walk starting at c should necessarily contain a in the second step. Observe that G does not contain the edge (e, f) (resp. (f, e)), for otherwise this edge, in combination with a path from f (resp. e) to $\{a, c\}$, would imply the independence of $\{e, f\}$. Moreover,

$\{e, c\}$ is a basis of $N(G)$. Then G necessarily contains edges of type (e, x) where $x \in \{c, d\}$, since otherwise a walk starting at e and a walk starting at c will either meet at a or move to two new positions $\{a, b\}$ (if $(e, b) \in E$). Since $\{a, b\}$ is a circuit, we get a contradiction with the independence of $\{e, c\}$. The same can also be deduced for f . Hence, we know that E contains at least an edge of type (e, x) for some $x \in \{c, d\}$ and at least an edge of type (f, z) for some $z \in \{c, d\}$. Suppose now that E contains also an edge of type (e, y) where $y \in \{a, b\}$ (resp. (f, w) where $w \in \{a, b\}$). Then starting at $\{e, f\}$ we can move to $\{y, z\}$ where $y \in \{a, b\}$ and $z \in \{c, d\}$ (resp. to $\{x, w\}$ where $x \in \{c, d\}$ and $w \in \{a, b\}$). This leads to the erroneous conclusion that the circuit $\{e, f\}$ has a basis $\{y, z\}$ (resp. $\{x, w\}$) among its successors. We can therefore say that all edges of $\delta^+(e)$ (resp. $\delta^+(f)$) are of type (e, x) (resp. (f, x)) where $x \in \{c, d\}$. This holds if the outdegree of c is exactly 1. Similarly, if we also assume that the outdegree of a is exactly 1, we get that the edges of $\delta^+(e)$ and $\delta^+(f)$ necessarily have an extremity in $\{a, b\}$. This leads again to contradiction with the previous conclusion. We can then assume that the outdegree of a is strictly larger than 1. As said in the beginning of the proof, E contains (a, x) and also (x, a) . Observe now that $x \notin \{a, c, e, f\}$ (using what we know about the edges of $\delta^+(e)$ and $\delta^+(f)$). Then, $x = d$ and the outdegree of a is exactly equal to 2. This implies that any walk starting at a will either move to c or to d . Now using the fact that the edges of $\delta^+(e)$ have their terminal extremity in $\{c, d\}$, we get that any pair of walks starting at $\{a, e\}$ will either meet at c , or d , or move to $\{c, d\}$ leading to another contradiction (since $\{c, d\}$ is a circuit and $\{a, e\}$ is a basis).

Case 2. Assume that G contains a cycle of length 2 in addition to at least one loop. Without generality loss, assume that E contains (a, c) and (c, a) . Loops should be around either a or c (otherwise, we get cycle subdigraph whose size is at least 3). Assume then that G contains at least the loop (c, c) . Let $(d, x) \in E$ be any edge. Observe that whatever x is, either $\{x, a\}$ or $\{x, c\}$ is a basis. Hence, the circuit $\{d, c\}$ has at least one successor that is a basis. This is of course impossible.

Case 3. Assume now that G does not contain cycles of size 2. Then G contains exactly two loops (x, x) and (y, y) . Notice that $\{x, y\}$ should be a basis. Without loss of generality we can assume that $x = a$ and $y = c$. Since $\{b\}$ is independent in $N(G)$, there should be at least a path from b to either a or c . Assume that there is a path P from b to c . If P does not go through a , then one can build a walk starting at a and staying there and a walk starting at b , moving to c through P and then staying at c . This contradicts the fact that $\{a, b\}$ is a circuit of the matroid. Assume now that P goes through a . Hence P contains a subpath P_1 connecting b to a and a subpath P_2 connecting a to c . One can again build a walk starting at b , going through P_1 and then staying at a , and a walk starting at a staying there until the first walk reaches a and then going through P_2 and staying at c . This contradicts again the fact that $\{a, b\}$ is a circuit. Consequently, G contains a path Q connecting b to a and there are no paths connecting b to c . Notice that this implies that there are no paths

connecting a to c . Similarly, there is a path Q' connecting d to c and there are no paths connecting d to a . Observe that Q and Q' are necessarily disjoint and there is no path from any vertex of Q to any vertex of Q' and vice-versa. Observe that $Q \cup Q'$ contains at least $\{a, b, c, d\}$. Notice that the outdegree of a is equal to 1 since any infinite walk starting at a and leaving a can not reach Q' so it should go back to a which is not possible due to lack of cycles and loops. The same holds for c .

Assume that E contains an edge of type (e, x) where x belongs to Q and an edge of type (f, y) where y belongs to Q' . Then one can obviously build two disjoint walks starting at e and f contradicting the circuit status of $\{e, f\}$. We can then assume without generality loss that all edges of type (e, x) satisfy $x \in V(Q)$ while all edges of type (f, y) satisfy $y \in V(Q) \cup \{e\}$. Observe that any infinite walk starting at e should reach a and stay there implying that $\{e, a\}$ is dependent which is again not possible. \square

Let us now recall the definition of gammoid. A gammoid is defined using a directed graph $G = (V, E)$, a subset of sources $S \subseteq V$ and a subset of sinks $T \subseteq V$. The ground set of the gammoid is S and a subset $I \subseteq S$ is independent if I can be linked to $|I|$ vertices of T through vertex-disjoint paths. The class of gammoids is minor-closed and dual-closed. It contains laminar matroids and transversal matroids. When there is no restriction related to S (i.e., $S = V$), we get the subclass of strict gammoids [7, 10].

Proposition 6.5. *Each no-meet matroid is a gammoid.*

Proof. Consider the multi-layered graph of Figure 1. The graph can be slightly modified by merging vertices v_i^1 and v_i^2 and eliminating the source s and the sink t . Consider then the gammoid defined using this graph where the set of sources S is the set of vertices of the first layer (B^1) while the set of sinks is the set of vertices of the last layer (B^L). There is here a one-to-one correspondence between independent sets of this gammoid and independent sets of $N(G)$. \square

Notice that Proposition 6.5 gives a more direct proof of the fact that no-meets are matroids. Let us now focus on the connection with strict gammoids.

Proposition 6.6. *There is a no-meet matroid that is not a strict gammoid.*

Proof. An example given in [11] (Fig 2.18) shows that there is a transversal matroid that is not a strict gammoid. The example is in fact the dual of the laminar (and graphic) matroid given in Figure 5. Since each transversal matroid is a no-meet, we have a no-meet that is not a strict gammoid. \square

Proposition 6.7. *There is a strict gammoid that is not a no-meet matroid.*

Proof. Consider again the graphic matroid of Figure 5. We already proved in Proposition 6.4 that this matroid is not a no-meet matroid. On the other hand, it is already proved in [11] (Fig 2.18) that this matroid is a strict gammoid. \square

Let us finally consider the minors of no-meet matroids:

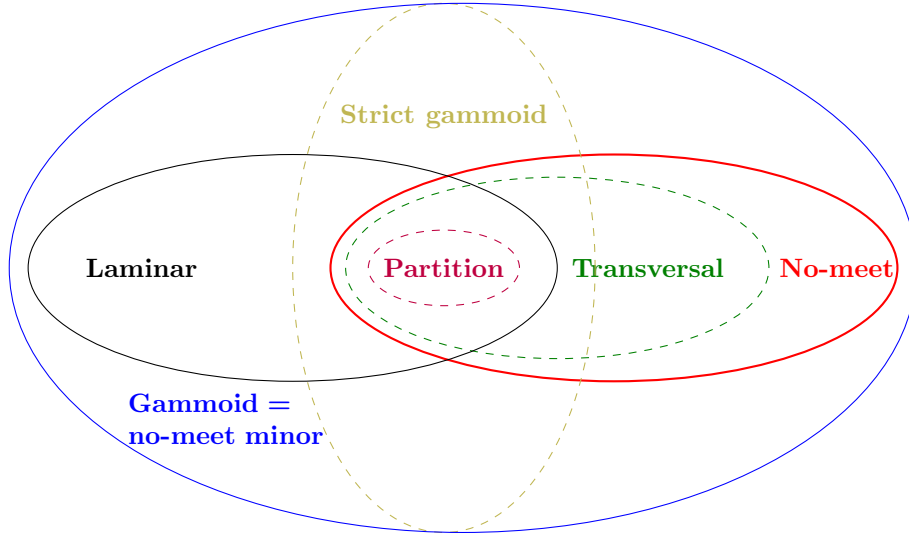


Figure 7: Connections of no-meet with other matroid classes

Proposition 6.8. *The class of no-meet minors is exactly the class of gammoids.*

Proof. Gammoids are known to be exactly the contractions of transversal matroids (see, e.g., [10], Corollary 39.5a). Combining with Theorem 6.1, we get that gammoids are minors of no-meet matroids. Moreover, since no-meet matroids are gammoids (Proposition 6.5), their minors are also minors of gammoids. Using that the class of gammoids is minor-closed, we deduce that all no-meet minors are gammoids. Combination of the two inclusions proves the wanted result. \square

Figure 7 summarizes the main connections presented in this section. Observe that the class of partition matroids is a subset of all mentioned classes of matroids. We could also add that no-meet matroids can be connected or disconnected. It is also easy to see that the direct sum of two no-meet matroids is a no-meet matroid (see [11] for definitions). Since no-meet matroids are gammoids, they are representable over all fields, except for a finite number of finite fields [13, 14].

7 Extensions and related applications

7.1 A no-meet edge version

One can naturally consider the variant where some vehicles placed on the graph nodes should move without having more than one robot/vehicle on the same edge. Notice that many vehicles can potentially be on the same node. This edge version can be easily transformed into the no-meet problem (the vertex version)

by considering the line digraph $L(G)$. Recall that given a digraph $G = (V, E)$, the line digraph of G has E as vertex set and an edge from $e_1 = (i_1, j_1) \in E$ to $e_2 = (i_2, j_2) \in E$ if and only if $j_1 = i_2$. Note that vertex-disjoint cycles in $L(G)$ correspond to edge-disjoint cycles in G . Consequently, the maximum number of walks that never meet in edges (or the maximum number of vehicles that we can place on the graph nodes) is equal to the size of the largest cycle subdigraph of $L(G)$.

We can also define a *no-meet edge matroid* denoted by $Ne(G)$. Given a digraph $G = (V, E)$, let $Ne(G) = (E, \mathcal{J})$ where a subset $F \subseteq E$ belongs to \mathcal{J} if and only if $\{v_e, e \in F\}$ is independent in $N(L(G))$ where v_e is the vertex of $L(G)$ corresponding to the edge $e \in E$.

Let us focus on the specific problem where we consider a set of vertices $S \subseteq V$ and a number n_i of vehicles placed in each vertex of $i \in S$. We would like to know whether it is possible for these vehicles to move forever without meeting in edges. Remember that each edge $(i, j) \in E$ is represented by a vertex $v_{(i,j)} \in V(L(G))$. The problem above is equivalent to check whether there is a set $S' \subset V(L(G))$ that is independent in $N(L(G))$ such that $|\{v_{(i,j)} \in S' : (i, j) \in E(G)\}| = n_i$ for each $i \in S$. One way to answer this question is to create a graph G'_S obtained from the line graph $L(G)$ by adding n_i copies of each vertex $i \in S$ and adding edges from each copy of i to all vertices of type $v_{(i,j)}$. The vehicles starting at S (n_i vehicles at each vertex $i \in S$) can move forever without meeting in edges, if and only if, the subset S'' containing the n_i copies of each vertex i of S is independent in $N(G'_S)$. Notice that G'_S and $L(G)$ have the same cycle subdigraphs since the vertices of S'' do not belong to cycles. Checking the independence of S'' can be done in polynomial time using the techniques of Section 3.

7.2 A capacitated version and related applications

A slightly more general version of the no-meet problem is obtained by considering limited meeting capacities for vertices. Say that each vertex $v \in V$ has a capacity c_v . Then we require the number of vehicles traversing vertex v at the same time to be less than or equal to c_v . Capacities are naturally assumed to be positive integer numbers. One can define an extended graph G' where c_v copies of each vertex $v \in V$ are considered and an edge from each copy of v to each copy of w is added whenever $(v, w) \in E(G)$. By considering the matroid $N(G')$, we get the wanted information about the acceptable starting positions with the number of vehicles at each position.

More generally, in addition to vertex meeting capacities, we can also consider edge meeting capacities c_e (the number of walks using edge e in the same time cannot be above c_e). We can also consider a specific length of the walks (L is equal to some fixed constant). This situation can still be easily modeled using a multi-layer graph that is similar to the graph G_S of Figure 1. We update the capacity of edges inside the graphs B^i (c_v for edge (v_i^1, v_i^2)) and edges from B^i to B^{i+1} (c_e for edge (v_i^2, w_{i+1}^1) where $e = (v, w)$). We might also consider lower bounds for the flow on edges of type (v_i^1, v_i^2) and edges of type (v_i^2, w_{i+1}^1) .

Vertex s can be connected to all vertices v_1^1 of B^1 . The capacity of the edges originating from s might be chosen depending on specific constraints related to the number of walks (vehicles) that can start at some vertex. The capacity of edges terminating at t are also updated in the same way. Any integer flow from s to t can be decomposed into walks satisfying the requirements.

A straightforward application of the model might be the following *coordinated multi-bus routing*. Given a set of routes, crossroads and depots, and k buses, we would like to design the itinerary of each bus such that, at each time slot and each route, the number of buses going through the route is higher than some lower bound (potentially depending on the time slot and the route) and lower than some upper bound (some kind of meeting capacity). Some limits are also considered for crossroads and depots. Buses should start at time slot 1 and stop at time slot L . Loops might also exist to model the fact that a bus might stay for sometime at some points (important bus stations, depots, etc.). If some routes need more time to be travelled, one can replace some edges by a sequence of consecutive edges. Weights can also be integrated to model some travelling cost. This coordinated multi-bus routing problem can then be solved by considering a minimum-cost flow problem where the flow value is fixed to k (the number of buses). The problem falls in the framework of vehicle routing problems (see, e.g., [15]). It can also be seen as a variant of the multiple chinese postman problem for which several integer programming formulations are generally proposed to integrate more specific constraints (see, e.g., [16] and the references therein).

7.3 Automata and FSMs related applications

Finite State Machines (FSMs) and Automata are widely used in modeling discrete event systems and in fact, allow to simulate the behavior of a system that moves from one state to another when an input is applied or an action is taken. FSMs are sometimes referred to as *automata with outputs* as an applied input at a current state requires an output to be produced. A transition diagram of an automaton is then represented by a graph where each state is represented by a vertex and each edge is labelled by an input i ; while in FSMs, an edge is labelled by an input/output pair i/o .

State identification of FSMs generally concerns the following problem: given an FSM, one should derive an input sequence $\alpha = i_1, \dots, i_k$ such that after its application and the observation of the output response (FSM reaction) $\beta = o_1, \dots, o_k$ the initial or the final/current FSM state can be uniquely identified (see, for example, [17, 18, 19] for more details). In automata, no output responses are considered, and thus their state identification focuses on the final state, namely an input sequence $\alpha = i_1, \dots, i_k$ should be derived, such that its application brings the automaton to a unique final state; this sequence is referred to as a *synchronizing sequence*.

We are then interested in checking the existence and deriving synchronizing sequences for finite automata and so called *distinguishing sequences* for FSMs. Note that even if the derivation of such sequences is well studied, the problem

remains of the interest of the community as the existence check of a distinguishing sequence is PSPACE-complete [18], even for deterministic FSMs; derivation of a shortest synchronizing sequence is NP-hard [19].

Since the observations can be highly corrupted, one might look for a *robust distinguishing sequence* that leads to sufficiently different observations. Assuming that S is the set of possible initial states and h is some chosen threshold, we refer to an input sequence α as an *h -robust distinguishing sequence*, if $\forall s_1, s_2 \in S, d(out(s_1, \alpha), out(s_2, \alpha)) > h$ where $out(s, \alpha)$ is the observed output sequence when we start at s and d is the Hamming distance (the number of different components between two sequences).

Consider the matroid $N(G)$ where G is the graph related to the FSM. If S is dependent, then any set of $|S|$ walks starting at S will meet after at most $O(n^5)$ steps. This means that there exists at least two states s_1 and s_2 such that $d(out(s_1, \alpha), out(s_2, \alpha)) = O(n^5)$. Therefore, an h -robust distinguishing sequence does not exist when h is chosen such that $n^5 = o(h)$.

Transition graphs with the set S which is dependent, have indeed interesting state identification properties for FSMs and automata. Similarly to the distinguishing sequence, we can consider the matroid $N(G)$ where G is the transition graph of an automaton. If each pair of vertices is dependent, $rank(S) = 1$, then each state pair of the automaton has a synchronizing (merging) sequence. If the automaton is complete and deterministic, then there always exists a synchronizing sequence for it.

Another property of independent state sets can be observed when considering a power automaton for a non-deterministic single-input automaton \mathbf{A} . The states of such power automaton are state subsets of \mathbf{A} , and there is a transition from S to S' (where $|S'| \leq |S|$) if S' is a successor of S (as defined in Section 1) in the digraph G representing the possible transitions of \mathbf{A} . Assume that this power automaton is a Büchi automaton where the final states are those corresponding to the subsets of size k and the initial state is any subset S_0 of size k . Consider again the matroid $N(G)$. In this case, the language of the constructed Büchi automaton is not empty, if and only if S_0 is independent in $N(G)$. Note here the remarkable fact that even if the number of states of the power automaton is exponential, the non-emptiness of its language can be checked in polynomial time.

8 Further questions

To conclude this paper, let us discuss some open questions that require further study.

We used the index of imprimitivity in Section 3 to show that the number of layers in the graph G_S of Figure 1 can be $O(n^3)$ for strongly-connected components and $O(n^5)$ for general graphs. It would be nice to give better bounds of the number of layers and to show that they are tight.

In fact, it is easy to see that in our proofs, we do not really need to have $A^{\otimes k+c} = A^{\otimes k}$ for all $k \geq k_{n,c}$ as announced in Lemma 3.1. It is enough to

have $A^{\otimes k+c} \geq A^{\otimes k}$ for k larger than some number. This leads to the following algebraic problem that might be studied for any Boolean matrix A :

$$\begin{cases} \min k + c \\ A^{\otimes k+c} \geq A^{\otimes k} \\ k \geq 1, c \geq 1, k \in \mathbb{N}, c \in \mathbb{N}. \end{cases}$$

Many problems related to no-meet mixed graph orientation deserve more study. What is, for example, the complexity of computing $r_H^{max}(S)$ and $r_H^{min}(S)$ when $|S|$ is bounded by a constant. Approximation algorithms with performance guarantees can also be proposed. Integer programming approaches for the NP-hard variants might also be developed.

Finally, the recognition problem of no-meet matroids still needs to be studied (given a matroid M , check whether it is a no-meet and find a digraph G such that $M = N(G)$). More structural properties of the no-meet matroids need to be explored and the exact intersection with other classes of matroids deserves further study.

Acknowledgments

We would like to thank an anonymous referee who raised interesting questions, helping the authors to better outline some of the topics.

References

- [1] A. Dessmark, P. Fraigniaud, D. R. Kowalski, A. Pelc, Deterministic rendezvous in graphs, *Algorithmica* 46 (1) (2006) 69–96.
- [2] N. H. Bshouty, L. Higham, J. Warpechowska-Gruca, Meeting times of random walks on graphs, *Inf. Process. Lett.* 69 (5) (1999) 259–265.
- [3] B. De Schutter, B. De Moor, On the sequence of consecutive powers of a matrix in a Boolean algebra, *SIAM Journal on Matrix Analysis and Applications* 21 (1) (1999) 328–354.
- [4] B. De Schutter, Upper bounds for the index of cyclicity of a matrix, Tech. Rep. 98-32, ESAT-SISTA, K.U.Leuven, Leuven, Belgium, revised version (Jul. 1999).
- [5] J. Bang-Jensen, G. Gutin, *Digraphs: Theory, Algorithms and Applications*, 2nd Edition, Springer-Verlag, London, 2009.
- [6] W. T. Tutte, A short proof of the factor theorem for finite graphs, *Canadian Journal of Mathematics* 6 (1954) 347–352.
- [7] S. Arora, C. Lund, R. Motwani, M. Sudan, M. Szegedy, Proof verification and the hardness of approximation problems, *J. ACM* 45 (3) (1998) 501–555.

- [8] R. Kohli, R. Krishnamurti, P. Mirchandani, The minimum satisfiability problem, *SIAM J. Discret. Math.* 7 (2) (1994) 275–283.
- [9] R. A. Brualdi, E. B. Scrimger, Exchange systems, matchings, and transversals, *Journal of Combinatorial Theory* 5 (3) (1968) 244–257.
- [10] A. Schrijver, *Combinatorial Optimization: Polyhedra and Efficiency.*, Springer, 2003.
- [11] J. G. Oxley, *Matroid Theory* (Oxford Graduate Texts in Mathematics), Oxford University Press, Inc., USA, 2006.
- [12] T. Fife, J. Oxley, Laminar matroids, *Eur. J. Comb.* 62 (C) (2017) 206–216.
- [13] J. H. Mason, On a Class of Matroids Arising From Paths in Graphs, *Proceedings of the London Mathematical Society* s3-25 (1) (1972) 55–74.
- [14] B. Lindström, On the vector representations of induced matroids, *Bulletin of the London Mathematical Society* 5 (1) (1973) 85–90.
- [15] K. Braekers, K. Ramaekers, I. Van Nieuwenhuyse, The vehicle routing problem: State of the art classification and review, *Computers & Industrial Engineering* 99 (2016) 300 – 313.
- [16] A. Shafahi, A. Haghani, Generalized maximum benefit multiple chinese postman problem, *Transportation Research Part C: Emerging Technologies* 55 (2015) 261 – 272.
- [17] E. F. Moore, Gedanken-experiments on sequential machines, in: *Automata Studies*, Princeton University Press, Princeton, NJ, 1956, pp. 129–153.
- [18] D. Lee, M. Yannakakis, Testing finite-state machines: State identification and verification, *IEEE Transactions on Computers* 43 (3) (1994) 306–320.
- [19] S. Sandberg, Homing and synchronizing sequences, in: *Model-Based Testing of Reactive Systems: Advanced Lectures*, Springer Berlin Heidelberg, 2005, pp. 5–33.