Sequence Graphs: Characterization and Counting of Admissible Elements

Sammy Khalife

Abstract We present a family of graphs implicitly involved in sequential models, which are obtained by adding edges between elements of a discrete sequence appearing simultaneously in a window of size w, and study their combinatorial properties. First, we study the conditions for a graph to be a sequence graph. Second, we provide, when possible, the number of sequences it represents. For $w = 2$, unweighted 2-sequence graphs are simply connected graphs, whereas unweighted 2-sequence digraphs form a less trivial family. The decision and counting for weighted 2-sequence graphs can be transformed by reduction into Eulerian graph problems. Finally, we present a polynomial time algorithm to decide if an undirected and unweighted graph has the said property for $w \geq 3$. The question of NP-hardness is left opened for other cases.

Keywords Graphs · Combinatorics · Representations

1 Introduction

The graphs we are interested in this paper, referred to as sequence graphs, represent the co-occurrences (potentially oriented) of the elements in a sequence appearing simultaneously in a window of constant size w. These structures encode information of several sequential models, in particular for natural language [4, 7, 9], supplementing the information of bag-of-words representations, which are invariant to any permutation. They also have been used for biological sequences, namely for protein visualization or protein-protein interaction prediction [2, 8]. In this work, we are interested in two main questions; first the question of recognition of such graphs, and second, the counting of corresponding sequences.

S. Khalife (✉)
LIX CNRS Ecole Polytechnique, Institut Polytechnique de Paris, Palaiseau, France
e-mail: khalife@lix.polytechnique.fr

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2021

C. Gentile et al. (eds.), *Graphs and Combinatorial Optimization: from Theory to Applications*, AIRO Springer Series 5,
https://doi.org/10.1007/978-3-030-63072-0_17

1.1 Definitions and Problem Statement

In the following, let $x = x_1, x_2, \ldots, x_p$ be a finite sequence of discrete elements among a finite vocabulary X. Without loss of generality, we can suppose that $X = \{1, \ldots, n\}$, let $I_p = \{1, \ldots, p\}$ and let $\mathbb{N}^*$ be the set of strictly positive integers.

Definition 1 $G = (V, E)$ is the graph of the sequence x with window size $w \in \mathbb{N}^*$ if and only if $V = \{x_i \mid i \in I_p\}$, and

$$(i, j) \in E \iff \exists (k, k') \in I_p^2,\ |k - k'| \leq w - 1,\ x_k = i \text{ and } x_{k'} = j \tag{1}$$

For digraphs, Eq. (1) is replaced by

$$(i, j) \in E \iff \exists (k, k') \in I_p^2,\ k \leq k' \leq k + w - 1,\ x_k = i \text{ and } x_{k'} = j \tag{2}$$

Finally, a weighted sequence digraph G is endowed with the matrix $\Pi(G) = (\pi_{ij})$ such that:

$$\pi_{ij} = \text{Card}\ \{(k, k') \in I_p^2 \mid k \leq k' \leq k + w - 1,\ x_k = i \text{ and } x_{k'} = j\} \tag{3}$$

By convention, a weighted (undirected) sequence graph is endowed with $\Pi = (\pi_{ij})$, $\pi_{ij} = \pi^{'}_{ij} + \pi^{'}_{ji}$ if $i \neq j$ and π'_{ij} otherwise, where π' verifies Eq. (3).

We say that x is a w-admissible sequence for G if G is the graph of the sequence x. G is referred to as the w-sequence graph of x with window size w.

π_{ij} represents the number of co-occurrences of i and j in a window of size w. Hence, the graph of a sequence x is unique for a given w. In the following, we use $G_w(x)$ as a shorthand for the w-sequence graph of x. In the weighted and directed case, it can be obtained with Algorithm 1.

Algorithm 1: Construction of a weighted sequence digraph

```
Data: Sequence $x$ of length $p$, window size $w$, $p \geq w \geq 2$
Result: $(G_w(x),\ \Pi)$
$V \leftarrow \emptyset$;
$d \leftarrow$ number of distinct elements of $x$;
Initialize $\Pi = (\pi_{i,j})$ to $d \times d$ matrix of zeros;
for $i = 1 \rightarrow p - 1$ do
    $V \leftarrow V \cup \{x_i, x_{i+1}\}$ ;
    for $j = i + 1 \rightarrow \min(i + w - 1, p)$ do
        $\pi_{x_i,x_j} \leftarrow \pi_{x_i,x_j} + 1$;
    end
end
Return $V,\ \Pi$
```

If G is not oriented, one should replace line 7 of Algorithm 1 by the "symmetrized" update:

$$\begin{array}{llll} \text{if } \pi_i \neq \pi_j : & \alpha \leftarrow \pi_{x_i,x_j}, & \pi_{x_i,x_j} \leftarrow \alpha + 1, & \pi_{x_j,x_i} \leftarrow \alpha + 1 \\ \text{else} : & \pi_{x_i,x_i} \leftarrow \pi_{x_i,x_i} + 1 & & \end{array} \tag{4}$$

The procedure in Algorithm 1 defines a correspondence between the sequence set S_X into the graph set $\mathscr{G}$: $\phi_w : S_X \to \mathscr{G}, x \mapsto G_w(x)$. $G \in \operatorname{Im} \phi_w$ exactly means that G is a $w-$sequence graph. For a given w, the two problems we address in this paper are the characterization (or recognition) of w-sequences graph, and the counting of the number of their w-admissible sequences.

1.2 Related Work

Despite their relations with co-occurrences based models for language [1, 7, 9], no such combinatorial questions were investigated in computational linguistics which we believe to be of interest, namely to understand the degree of ambiguity of these models. Besides, such structures have been partially studied in the Distance Geometry (DG) literature before, mostly to do with proteins, where an "atom window" can be defined by using the protein backbone [6]. However, the type of graph studied in Distance geometry does not refer directly to the results we are investigating in this paper. Indeed, the necessary and sufficient conditions for which such study would apply are:

- each element of the sequence x is associated with a unique vertex (which is not the case we investigate here, since a symbol can be repeated several times but only one vertex is created)
- the absence of loops

As a consequence, the results mentioned in the DG survey [6] do not apply to the present case.

1.3 Notations

In the following, we use $\mathscr{M}_d(\mathbb{N})$ as a shorthand for the square $d \times d$ matrices over the set of natural integers, $\operatorname{Tr}(M)$ for the trace of a matrix M, and $\operatorname{Sp}(M)$ for its set of eigenvalues.

2 2-Sequence Graphs

In this section, we consider $w = 2$. Algorithm 1 encodes each adjacency in the sequence x as an edge in $G_w(x)$. Obviously, the simplest case concerns undirected graphs as stated in the:

Proposition 1 *Let $G = (V, E)$ be an unweighted and undirected graph with $|V| > 1$. Then, the following assertions are equivalent:*

(i) G is connected
(ii) G has a 2-admissible sequence
(iii) G admits an infinite number of 2-admissible sequences

Proof If G is connected, a sequence is obtained by visiting all edges, for instance using a list of arbitrary sequences and shortest paths. The other implications are immediate. □

For digraphs, the previous characterization is wrong, even with strong connectivity. A counter example is given in Fig. 1. However, strong connectivity remains a sufficient condition:

Proposition 2 *Let $G = (V, E)$ be an unweighted digraph. If G is strongly connected then $G \in \operatorname{Im} \phi_2$. Moreover, a 2-admissible sequence can start or end at any given vertex of G.*

Proof Straightforward, similarly to (i) $\Longrightarrow$ (ii) for Proposition 1. □

Proposition 3 *Let $G = (V, E)$ be an unweighted digraph. If G is Eulerian or semi-Eulerian, then $G \in \operatorname{Im} \phi_2$.*

Proof If G is Eulerian or semi-Eulerian, there exists a walk going through all edges, this walk defines a 2-admissible sequence. □

Again the converse of Proposition 3 does not hold as depicted in Fig. 2. First, it is natural to consider the case of directed acyclic graphs (DAGs):

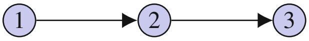

Fig. 1 *G* has 1 2 3 as a 2-admissible sequence but is not strongly connected

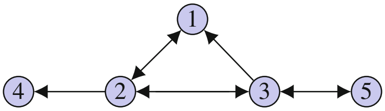

Fig. 2 *G* has 3 5 3 1 2 1 2 3 2 4 as a 2-admissible sequence but is not Eulerian nor semi-Eulerian

Proposition 4 *Let $G = (V, E)$ be a DAG. G is a 2-sequence graph if and only if it is a directed path, i.e. G is a directed tree where each node has at most one child and at most one parent. In this case, G has a unique 2-admissible sequence.*

Proof If G is a directed path, since G is finite, it admits a source node. Therefore a 2-admissible sequence is obtained by simply going through all vertices from the source node. This is obviously the only one.

Conversely, let us suppose G is a DAG and a 2-sequence graph. If G is not a directed path, there are two cases: either there exists a vertex having two children, or two parents. Let s be a vertex having 2 distinct children c_1 and c_2. This is not possible since there cannot be a walk going through (s, c_1) and (s, c_2): G would have a cycle otherwise. Finally a vertex v cannot have two parents p_1 and p_2: if a 2-admissible sequence existed, it would have to go through (p_1, v) and (p_2, v), creating a cycle, hence the contradiction. □

Every directed graph G is a DAG of its strongly connected components. In the following, let $R(G)$ be the DAG obtained by contracting the strongly connected components of G.

Proposition 5 *Let $G = (V, E)$ be a digraph. If G is a 2-sequence graph then $R(G)$ is a 2-sequence graph.*

Proof Let G be a 2-sequence graph, and let us suppose that $R(G)$ is not a 2-sequence graph. Since $R(G)$ is a (weakly) connected DAG, then using Proposition 4, it cannot be a directed path, so $R(G)$ has either a node having two children or two parents. Let S be a node of $R(G)$ having at least 2 distinct children C_1 and C_2. This means that there exist three distinct corresponding nodes in V, s, v_1 and v_2 such that $(s, v_1) \in E$ and $(s, v_2) \in E$. Since G is a 2-sequence graph, there exists a walk covering (s, v_1) and (s, v_2), such walk would make S, C_1 and C_2 the same node in $H(G)$, hence the contradiction. The case for which a vertex has two parents is dealt with similarly. □

The converse of Proposition 5 does not hold as depicted in Fig. 3, which motivates the following definition.

Definition 2 Let G be a digraph, and $R^+(G)$ be the weighted DAG obtained from $R(G)$, such that the weight of an edge is the number of distinct arcs from two strongly connected components in G.

Theorem 1 *Let $G = (V, E)$ be an unweighted digraph.*

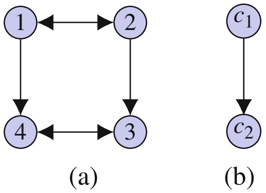

Fig. 3 G is not a 2-sequence graph while $R(G)$ is. (**a**) G. (**b**) $R(G)$

G is a 2*-sequence graph if and only if $R^+(G)$ is a directed path and its weights are all equal to* 1.

Proof If G is a 2-sequence graph, $R(G)$ is a 2-sequence graph using Proposition 5. Also Proposition 4 implies that $R(G)$ and $R^+(G)$ are directed paths. Moreover, if $R^+(G)$ had a weight strictly greater than 1, then there would be strictly more than one edge between two strongly connected components C_1 and C_2. All these edges go in the same direction otherwise $C_1 \cup C_2$ would be part of a larger strongly connected component. This is a contradiction since any 2-admissible sequence would have to go from C_1 to C_2 and then come back to C_1 (or conversely) and $C_1 \cup C_2$ would again be part of a larger strongly connected component.

Conversely, let us suppose $R^+(G)$ is a a directed path and its weights are equal to one. First, there exists a walk $x_1, \dots, x_p$ covering all edges of $R^+(G)$ verifying: (i) $\forall i,\ x_i \in V$ or x_i represents a strongly connected component of G, (ii) there is only one edge in G between from x_i to x_{i+1} and (iii) x has no repetition, i.e. there is no common vertex in G between x_i and x_{i+1}. We construct a 2-admissible sequence y for G by means of the following procedure.

Initialisation: If $x_1 \in V$, we simply set $y \leftarrow x_1$. Otherwise, x_1 corresponds to a strongly connected component C_1 of G and we add to y any 2-admissible sequence of C_1.

For $i \in \{1, .., p-1\}$:

- If $(x_i, x_{i+1}) \in E$: we add x_{i+1} to the sequence y.
- If $x_i \in V$ and x_{i+1} is a strongly connected component C_i of G: By assumption, there exists only one edge of G from x_i to a vertex of C_i, say c_0^i. Since C_i is strongly connected, using Proposition 2, C_i has a walk going through all of its edges and starting in c_0^i, say $c_0^i, \dots, c_p^i$. We add $c_0^i, \dots, c_p^i$ to y.
- If x_i corresponds to a strongly connected component C_i and $x_{i+1} \in V$: we perform similar operations by stopping on the single node of C_i that has a edge to x_{i+1} (this is possible thanks to Proposition 2).
- x_i and x_{i+1} both correspond to strongly connected components C_i and C_{i+1}, there exists only one edge between in E between C_i and C_{i+1}, say $e_i = (v_i, v_{i+1})$. We can complete y by a walk from the last vertex visited which belong to C_i and v_i, and then by a 2-admissible sequence through C_{i+1} starting in v_i and ending in v_{i+1}.

The process stops when $i = p - 1$, and all edges are covered by the sequence y. □

Therefore, an algorithm to decide if a digraph is a 2-sequence graph is obtained by extracting its strongly connected components (there exist linear time algorithms e.g. [10]), and to count the number of distinct edges between these.

Corollary 1 *Let G be an unweighted digraph. The possible numbers of* 2*-admissible sequences for G is exactly* $\{0, 1, +\infty\}$. *Moreover, G admits a unique* 2*-admissible sequence if and only if G is a directed path.*

Proof Let G a be 2-sequence graph. G verifies the characterization of Theorem 1. If $R(G)$ has a vertex C representing a strongly connected component of G (or

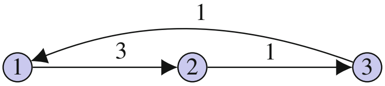

Fig. 4 *G* is strongly connected but is not a 2-sequence graph

a vertex with a loop), then by adding an arbitrary number of cycles in C to the admissible sequence y (cf. Proof 2), the new sequence is still admissible. Otherwise, if every vertex of $R(G)$ is in V without self-loops in E, then G is a DAG. Using Proposition 4, y is the unique 2-admissible sequence. □

2.1 Weighted 2-Sequence Graphs

The weighted case cannot be treated similarly due to the constraint 3. A counterexample is depicted in Fig. 4. Moreover, a weighted graph has a finite number of admissible sequences. This property can be seen using Proposition 6 below.

Proposition 6 *If a graph is a weighted w-sequence graph, all of its admissible sequences have the same length.*

Proof Let x be a w-admissible sequence for G of length p. If G is a digraph, Algorithm 1 is incrementing $(p - w + 1)(w - 1) + \frac{(w-1)(w-2)}{2}$ times the total weight, therefore:

$$\sum_{i,j} \pi_{ij} = (p - w + 1)(w - 1) + \frac{(w-1)(w-2)}{2} \tag{5}$$

If $w \geq 2$, this yields: $p = w - 1 - \frac{w-2}{2} + \frac{1}{(w-1)} \sum_{i,j} \pi_{ij}$

Otherwise, if G is undirected, the weights matrix obtained with Algorithm 1 does not yield Eq. (5), due to the update of Eq. (4). The weights on the diagonal remain the same, but the others are multiplied by 2, hence the formula:

$$\sum_{i,j} \pi_{ij} + \mathrm{Tr}(\Pi) = 2(p - w + 1)(w - 1) + (w - 1)(w - 2) \tag{6}$$

leading to $p = \frac{1}{2(w-1)}[\sum_{i,j} \pi_{ij} + \mathrm{Tr}(\Pi)]$. □

Corollary 2 *Let G be a weighted w-sequence digraph, and Π its weights matrix. If w even, then $(w - 1) \mid \sum_{i,j} \pi_{ij}$.*

Corollary 3 *Let G be a w-sequence (undirected) graph and Π its weights matrix. Then $2(w - 1) \mid \sum_{i,j} \pi_{ij} + \mathrm{Tr}(\Pi)$.*

Definition 3 Let $\psi(G)$ be the auxiliary multigraph with the same vertices as $G = (V, E)$ and with π_{ij} edges between $(i, j) \in V^2$.

Due to the previous study, the characterization of weighted 2-sequence graphs using $\psi(G)$ is immediate. A semi-Eulerian graph is a graph that admits a Eulerian walk (instead of cycle for Eulerian graphs).

Theorem 2 *If G is a weighted graph (directed or not), with $\Pi(G) \in \mathscr{M}_d(\mathbb{N})$, then: $G \in \operatorname{Im}\phi_2 \iff \psi(G)$ is connected and semi-Eulerian.*

Proof $G \in \operatorname{Im}\phi_2$ means that there is a trail going through each edge $(i, j) \in E$ exactly π_{ij} times. This trail corresponds to a semi-Eulerian path in $\psi(G)$. □

2.2 Counting 2-Admissible Sequences for Weighted Graphs

Proposition 7 sums up the results for the counting problem of a weighted graph:

Proposition 7 *Counting the number of 2-sequences for a weighted graph is #P-complete. However, if G is a weighted digraph with $\Pi(G) \in \mathscr{M}_d(\mathbb{N})$, then the number p_2 of 2-admissible sequences is given by:*

$$p_2 = \frac{t(\psi(G))}{\prod_{e\in E}\pi_e!}\prod_{v\in V}\big(\deg_{\psi(G)}(\psi(v)) - 1\big)! \tag{7}$$

where $t(G)$ is the number of spanning trees of a graph G. If L is the Laplacian matrix of G, then $t(G)$ is given by $t(G) = \prod_{\substack{\lambda_i\in \mathrm{Sp}(L)\\ \lambda_i\neq 0}}\lambda_i$.

Proof Given a 2-admissible sequence of G, the choice of a corresponding Eulerian path in $\psi(G)$ is the choice of $\sigma = (\tau_1, \dots, \tau_{|E|})$ of $|E|$ permutations of $\{1, \dots, \pi_e\}$ representing the visit order in $\psi(G)$. $G \mapsto \psi(G)$ being bijective, counting Eulerian paths in an undirected graph is #P-complete [3], hence so is the problem of counting the 2-sequences of a weighted graph. BEST [11] and Matrix tree [5] theorems allow to derive formula (7) which guarantees in that the problem on digraphs is in P. □

To use formula (7), $\deg_{\psi(G)}(\psi(v))$ can be obtained using the following formula: $deg_{\psi(G)}(\psi(v)) = \sum_{n\in V}\pi_{nv} + \sum_{n\in V}\pi_{vn}$.

The results are summed up in Table 1.

Table 1 Results for various instances of our problems ($w = 2$)

	Undirected		Directed	
Problem	Unweighted	Weighted	Unweighted	Weighted
Nb. sequences	(P) $\{0, +\infty\}$	#P-hard	(P) $\{0, 1, +\infty\}$	(P) BEST Theorem
$G \in \operatorname{Im}\phi_2$?	G connected	$\psi(G)$ Eulerian or semi Eulerian	Theorem 1	$\psi(G)$ Eulerian or semi Eulerian

3 What Happens If $w > 2$?

The characterization of 3-graphs is not the same as for 2-graphs, as the counter-example in Fig. 5a shows: the depicted graph has no loop so there must at least one clique of size 3, which is not the case. Similarly, Fig. 5b depicts a counter example for directed graphs: G does not have loop, so if it had a 3-admissible sequence, such sequence must be of the form $\{1\,2\,3\,1\ldots, 1\,3\,2\,1\ldots, 2\,3\,1\,2\ldots, 3\,2\,1\,3\ldots, 2\,1\,3\,2\ldots\}$ but then (2, 1) would form an edge.

Similarly to the procedure in Sect. 2.1, we will use an auxiliary graph built on G. Let $H(G) = (E, E_H)$ be the new graph obtained with the following procedure. Two edges $e = (v_1, v_2)$, $f = (v_3, v_4)$ of E are connected in $H(G)$ if and only if (An illustration is given Fig. 6):

$$v_2 = v_3 \text{ and } (v_1, v_4) \in E \tag{8}$$

Therefore, by definition, a walk P in $H(G)$ is always of the form:

$$P = (t_1, t_2), \ldots, (t_{p-1}, t_p) \text{ s.t } \forall i \in \{1, \ldots, p-1\},\ (t_i, t_{i+1}) \in E \tag{9}$$

It is clear that if $H(G)$ is a 2-graph, then G is a 3-graph since there is a walk going through all edges of $H(G)$. However, the converse is not true as depicted in Fig. 7. In order to determine if $G = (V, E)$ has an admissible sequence for any w, a procedure is to recursively merge pairs of vertices, maintaining constraints defined below. These constraints are similar to Eq. (8). We adopt the following notations, $u_{i,j} = (u_i, u_j)$ and $u_{1:k} = (u_1, \ldots, u_k)$. The iterative procedure (for $w \geq 3$) is summed up in 10.

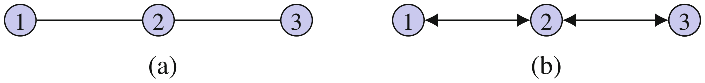

Fig. 5 Counter-examples for $w = 3$. (**a**) G is connected but does not have any 3-admissible sequence. (**b**) G is strongly connected but does not have any 3-admissible sequence

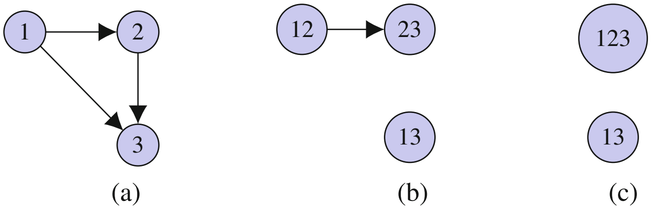

Fig. 6 Reduction on a simple example ($w = 3$). (**a**) Original graph G. (**b**) Graph H . (**c**) DAG $R(H)$

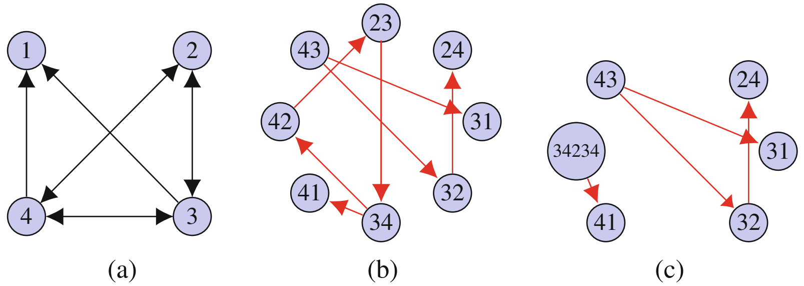

Fig. 7 Procedure to find a 3-admissible sequence. 34234, 41: is 3-admissible, with authentic sequence 3 4 2 3 4 1. (**a**) Original graph G. (**b**) Graph H is not a 2-sequence graph. (**c**) DAG $R(H^{(1)})$

Namely, $\forall k \in \{2, \ldots, w-2\}$, one has

$$E^{(k)} = \{u_{1:k+1} \in V^{k+1} \mid u_{1:k} \in E^{(k-1)}, u_{2:k+1} \in E^{(k-1)} \wedge (u_1, u_{k+1}) \in E\} \tag{10}$$

Let $H^{(k)} = (E^{(k)}, E^{(k+1)})$, it can be defined recursively through:

$$H^{(0)} = G \qquad\qquad \forall k \in \mathbb{N}^*, \quad H^{(k)} = f(H^{(k-1)}) \tag{11}$$

where f transforms edges into vertices and creates edges between new vertices that verify Eq. (10). It should be noted that $H(G)$ is directed if and only if G is.

Definition 4 Let u be a vertex of $H^{(k)}$ for $k \in \mathbb{N}$, $u = (u_1, \ldots, u_k, u_{k+1})$, where $u_j \in V$ for each j. The sequence $u_1, \ldots, u_{k+1}$ is the authentic sequence of u. We also call an authentic sequence of a walk on $H^{(k)}$: $P = (x_1, \ldots, x_{k+1}), (x_2, \ldots, x_{k+2}), \ldots, (x_v, \ldots, x_{v+k})$ the sequence $x_1, x_2, \ldots, x_{v+k}$.

In order to obtain admissible sequences of length p, the computation of $H^{(p)}$ requires p iterations, and the number of vertices and edges of $H^{(k)}$ can increase during iterations (the complete graph is an example for which theses numbers increase quadratically).

Proposition 8 *Let $x = x_1, \ldots, x_p$ be a w-admissible sequence of a graph (or digraph) $G = (V, E)$. If $w \leq p$, then x is an authentic sequence of a walk of length $p - w + 1$ on $H^{(w-2)}$.*

Proof Let $x = x_1, \ldots, x_p$ be a w-admissible sequence of G. Let P be a walk on $H^{(w-2)}$, and $P[i]$ be the i-th element of P, $P[i] \in H^{(w-2)}$: $P[i] = (P[i]_1, \ldots, P[i]_{w-1})$.

Let us suppose that $w \leq p$ (which we can always do), and let us show the following property by induction on k:

$$\begin{aligned}&\forall k \in \{w-1, \ldots, p\},\ \exists \text{ walk } P \text{ on } H^{(w-2)},\\ &\quad x_{1:k} = P[1]_1, P[2]_1, \ldots, P[k-(w-1)]_1, P[k+1-(w-1)]_{1:(w-1)}\end{aligned} \tag{12}$$

- Initialisation: $k = w - 1$. By construction of $H^{(w-2)}$, $x_{1:w-1}$ is the authentic sequence of "static walk": $P = P[1] = x_{1:w-1} \in H^{(w-2)}$.
- Induction: let us suppose the property is verified for $k \in \{w-1, \dots, p-1\}$, i.e. there exists a walk P on $H^{(w-2)}$ such that:

$$x_{1:k} = P[1]_1, P[2]_1, \dots, P[k-(w-1)]_1, P[k+1-(w-1)]_{1:(w-1)}$$

Since x is w-admissible, then by definition:

$$\forall i \in \{k+1-(w-1), \dots, k\}, \forall j \in \{i+1, \dots, \min\{k+1, i+w-1\}\} : (x_i, x_j) \in E$$

Therefore, by definition of $H^{(w-2)}$, $\xi^{k+1} = x_{k+1-(w-1)}, \dots, x_{k+1} \in H^{(w-2)}$.

Let $P[k+2-(w-1)] \mathrel{\hat{=}} \xi^{k+1}$, then $P[k+2-(w-1)]_{1:(w-1)} = x_{k+1-(w-1)}, \dots, x_{k+1}$. Besides, from the induction assumption: $\forall i \in \{1, \dots, k-(w-1)\}$, $P[i]_1 = x_i$. This ensures that: $x_{1:(k+1)} = P[1]_1, P[2]_1, \dots, P[k+1-(w-1)]_1, P[k+2-(w-1)]_{1:(w-1)}$ which ends the induction and the proof. □

Theorem 3 *Let G be a graph and $w \in \mathbb{N}^* - \{1, 2\}$. If G is undirected and unweighted then deciding if G is a w-sequence graph is in P.*

Proof It is possible to compute the connected components of $H^{(w-2)}$, say $C_1, \dots, C_m$, in polynomial time. For each $i \in \{1, \dots, m\}$, it is possible to construct walks covering all edges in polynomial time (for instance iteratively using shortest paths). Let $W_1, \dots, W_m$ be such walks and $X_1, \dots, X_m$ their respective authentic sequences. Using Proposition 8, G is a w-sequence graph if and only if there exists a walk $\tilde{W_{i_0}}$ on some C_{i_0} creating exactly the edges of G. However, W_{i_0} creates more edges than any walk on C_{i_0} by construction.

In conclusion, the assertion: $\exists i \in \{1, \dots, m\}$, $\phi_w(X_i) = G$ is a characterization of G being a w-sequence graph. This assertion is decidable in polynomial time since for all i, computing $\phi_w(X_i)$ requires a polynomial number of operations. □

For digraphs, the analogue of the aforementioned procedure would consist in enumerating all paths in the DAG $R(H^{(w-2)})$. However, the number of paths can be exponential, even for a sequence graph. For the sake of completeness, we will prove that the reduction by strongly connected components preserves admissibility.

Lemma 1 *Let x be a walk on $H^{(w-2)}$ whose authentic sequence is w-admissible for its corresponding unweighted graph G. If x goes through a strongly component C of $H^{(w-2)}$, adding any supplementary path of C to x lets x w-admissible. Any graph generated by a walk on $H^{(w-2)}$ can be generated by a walk on $R(H^{(w-2)})$.*

Proof Let $P = P[1], , \dots, P[r]$ be a walk on $H^{(w-2)}$ going through a strongly connected component C, with an arbitrary ordering of its vertices, i.e. $C = \{c_1, \dots, c_m\}$. This means $\exists (m_0, i_0) \in \{1, \dots, m\} \times \{1, \dots, r-1\}$ s.t $P[i_0] = c_{m_0}$ and $(c_{m_0}, P[i_0+1]) \in E$. Let $\mathscr{C} = c_{m_0}, c_{j_1}, \dots, c_{j_v}$ be a path in C with $(c_{j_v}, P[i_0+1]) \in E$. Let Q be the new path: $Q = P[1], \dots, P[i_0], c_{j_1}, \dots, c_{j_v}, P[i_0+$

Algorithm 2: A recognition algorithm for unweighted digraphs

```
Data: Graph $G$, window width $w$
Result: (Boolean, empty set or $w$-admissible sequence)
Build $H^{(w-2)}$ recursively (e.g. with 11);
Construct $R_H^w = R(H^{(w-2)})$ ;
for source-sink path of $R_H^w$ do
    if authentic sequence of path is $w$-admissible for $G$ then
        return (True, sequence)
    end
end
return (False, ∅);
```

1], ..., $P[r]$. By construction of $H^{(w-2)}$, the edges created by any walk on $H^{(w-2)}$ are in E, so Q is still admissible.

Let us label every node of $R(H^{(w-2)})$ representing a strongly connected component of $H^{(w-2)}$ by any $2-$admissible sequence (one exists thanks to Proposition 2). A walk on $H^{(w-2)}$: $x_1, \dots, x_p$ can be met by a walk on $R(H^{(w-2)})$ using the following procedure:

For $i \in \{1, \dots, p-1\}$:

- if $x_i, x_{i+1} \in E$, we keep x_i and x_{i+1}
- if $x_i \in V$ and x_{i+1} is in a strongly connected component of $H^{(w-2)}$ (but a node of $R(H^{(w-2)})$), represented by $c_1, \dots, c_{C_i}$, then a path from x_{i+1} to c_1 exists since the component is strongly connected: $x_{i+1}, p_1, \dots, p_m, c_1$. We keep x_i, x_{i+1}, $p_1, \dots, p_m, c_1, \dots, c_{C_i}$. Using the aforementioned result, this does not perturb admissibility.
- if $x_{i+1} \in V$ and x_i is in a strongly connected component of H^{w-2}, we proceed similarly (x_i and x_{i+1} are swapped).
- if both x_{i+1} and x_i are strongly connected components of H^{w-2}, we add intermediary nodes to connected both components similarly.

□

4 Conclusion

In this preliminary study, we considered two main combinatorial problems: the recognition problem of sequences graphs, and the counting of their realizations. Solving the second problem totally solves the first one, but in the trivial case $w = 2$, the first one is “simpler”: the recognition problem of sequence graphs is P for $w = 2$ for any data instance, but the counting problem is $\#P$-hard for weighted graphs. This justifies the distinction of these problems from a computational point of view.

Furthermore, for $w > 2$, the recognition problem is in P for one configuration (unweighted graphs), but the complexity classes of the other instances are left opened, and so are the counting problems for $w > 3$. A possible lead to answer these questions would be to investigate forbidden patterns in a sequence graph. Finally, it should be noted that the abstraction of sequences graphs exactly coincides with the graphs implicitly involved in co-occurrence models or point wise-mutual information models [1, 7, 9], used as input of algorithms to construct word representations. In these models, representations are ambiguous if the given weighted graph has several realizations. Therefore, other extensions of this work would be to propose scalable algorithms (or at least, for reasonable values of w and length of the sequences) to count and explicit realizations, in order to obtain more information about the degree of ambiguity in these models.

References

1. Arora, S., Li, Y., Liang, Y., Ma, T., Risteski, A.: A latent variable model approach to PMI-based word embeddings. Trans. Assoc. Comput. Linguist. **4**, 385–399 (2016)
2. Asgari, E., Mofrad, M.R.: Continuous distributed representation of biological sequences for deep proteomics and genomics. PLOS One **10**(11) (2015)
3. Brightwell, G., Winkler, P.: Counting Eulerian circuits is #p-complete. In: Proceedings of the Second Workshop on Analytic Algorithmics and Combinatorics (2005)
4. Broder, A.Z., Glassman, S.C., Manasse, M.S., Zweig, G.: Syntactic clustering of the web. Comput. Netw. ISDN Syst. **29**(8–13), 1157–1166 (1997)
5. Chaiken, S.: A combinatorial proof of the all minors matrix tree theorem. SIAM J. Algebraic Discrete Methods **3**(3), 319–329 (1982)
6. Liberti, L., Lavor, C., Maculan, N., Mucherino, A.: Euclidean distance geometry and applications. SIAM Rev. **56**(1), 3–69 (2014)
7. Mikolov, T., Yih, W.T., Zweig, G.: Linguistic regularities in continuous space word representations. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 746–751 (2013)
8. Ng, P.: dna2vec: Consistent vector representations of variable-length k-mers. arXiv preprint arXiv:1701.06279 (2017)
9. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014)
10. Sharir, M.: A strong-connectivity algorithm and its applications in data flow analysis. Comput. Math. Appl. **7**(1), 67–72 (1981)
11. van Aardenne-Ehrenfest, T., de Bruijn, N.: Circuits and trees in oriented linear graphs. In: Classic Papers in Combinatorics, pp. 149–163. Springer, Berlin (2009)