Properties of graphs specified by a regular language

Volker Diekert · Henning Fernau · Petra Wolf

Abstract
Traditionally, graph algorithms get a single graph as input, and then they should decide if this graph satisfies a certain property $\Phi$. What happens if this question is modified in a way that we get a possibly infinite family of graphs as an input, and the question is if there is a graph satisfying $\Phi$ in the family? We approach this question by using formal languages for specifying families of graphs, in particular by regular sets of words. We show that certain graph properties can be decided by studying the syntactic monoid of the specification language $L$ if a certain torsion condition is satisfied. This condition holds trivially if $L$ is regular. More specifically, we use a natural binary encoding of finite graphs over a binary alphabet $\Sigma$, and we define a regular set $G \subseteq \Sigma^*$ such that every nonempty word $w \in G$ defines a finite and nonempty graph. Also, graph properties can then be syntactically defined as languages over $\Sigma$. Then, we ask whether the automaton $A$ specifies some graph satisfying a certain property $\Phi$. Our structural results show that we can answer this question for all “typical” graph properties. In order to show our results, we split $L$ into a finite union of subsets and every subset of this union defines in a natural way a single finite graph $F$ where some edges and vertices are marked. The marked graph in turn defines an infinite graph $F^\infty$ and therefore the family of finite subgraphs of $F^\infty$ where $F$ appears as an induced subgraph. This yields a geometric description of all graphs specified by $L$ based on splitting $L$ into finitely many pieces; then using the notion of graph retraction, we obtain an easily understandable description of the graphs in each piece.

Research supported by DFG project FE 560/9-1.

Preamble: We dedicate the paper to Klaus–Jörn Lange on the occasion of his 70th birthday. The conference abstract of the present paper appeared in [80]. Here, we give full proofs and we correct some mistakes.

Volker Diekert
diekert@fmi.uni-stuttgart.de

Henning Fernau
fernau@informatik.uni-trier.de

Petra Wolf
wolfp@informatik.uni-trier.de

1 Formal Methods in Informatics, Universität Stuttgart, Stuttgart, Germany
2 FB 4 – Informatikwissenschaften, Universität Trier, Trier, Germany
1 Introduction

The paper is about families of finite graphs specified by regular languages, and their properties. When dealing with algorithms, a graph is often specified by its adjacency matrix or by its induced edge-list together with the number of isolated vertices, if there are any. In either representation, a graph comes with a linear order on the vertices and the edges are directed. Moreover, an adjacency matrix ignores multiple edges, but self-loops may occur. We follow these conventions in our paper. We encode a finite graph $G = (V, E)$ as a word over the binary alphabet $\Sigma = \{a, b\}$ as follows: The $i$th vertex $u_i$ of a graph is encoded by $ab^i$ and the edge $(u_i, u_j)$ is encoded by $ab^iaaab^ja$. Thus, every word $w$ in $\mathbb{G} = (ab^+a \cup ab^+aaab^+a)^+$ represents in a natural way a unique graph $\rho(w)$, because $ab^+a \cup ab^+aaab^+a$ is a regular code. Namely, $\rho(w)$ is the graph consisting of all vertices and edges whose encodings appear as a factor in $w \in \mathbb{G}$. Notice that it does not matter if a factor appears once or many times. Given a finite graph $G = (V, E)$ with any linear order on the vertices, we obtain a code word $\gamma(G)$ in $\mathbb{G}$ as follows. We write $V$ as $\{1, \ldots, |V|\}$ using the bijection induced by the linear order on $V$, and then we write the edges and the isolated vertices in the order which yields the short-lex normal form of $G$ in $\rho^{-1}(G)$. This means that first, all edges are listed and then, possible isolated vertices follow. We are interested in abstract graphs, only. Thus, isomorphic graphs are treated as equal. Therefore, several $\gamma(G)$’s are possible, depending on the linear orders on $V$, but even then, the short-lex normal form would give a unique syntactic representation of $G$ if necessary.

We cannot avoid that every nonempty graph $G$ has infinitely many representations $w \in \mathbb{G}$ such that $\rho(w) = G$. For example, the one-point graph $([\ast], \emptyset)$ is represented by all words in the regular set $L_i = (ab^i a)^+$ as soon as $i \geq 1$, i.e., for all $w \in L_i$ we have $\rho(w) = G$.

Given any $L \subseteq \mathbb{G}$, it defines a set of graphs $\rho(L)$. The main interest is when $\rho(L)$ is infinite but $L \subseteq \mathbb{G}$ is regular. The aim is to “understand” the infinite set of graphs in $\rho(L)$. It is far from obvious that this is possible. If $L$ is finite, then $\rho(L)$ is finite, too. But the converse is false. As we will see, if $L$ is regular, then we can decide finiteness of $\rho(L)$; and if $\rho(L)$ is finite, then we can compute all its graphs. However, if $\rho(L)$ is infinite, then a global understanding of $\rho(L)$ is, a priori, not easy.

1.1 A sketch of our approach and our results

Let us try to give a high-level explanation of the underlying geometric idea how to approach the family of graphs $\rho(L)$. Remotely, it is like understanding the geometry of a topological manifold using the fact that it locally resembles an Euclidean space. For example, it is possible to realize a torus (which is a compact two-dimensional surface) as a unit square nor into a soup tureen with two or more handles. In our case, we deal with purely combinatorial objects. Nevertheless, we wish to understand the set of graphs $\rho(L)$ by constructing a finite subset of graphs together with an “open neighborhood” around these graphs such that $\rho(L)$ is covered by that construction. Thus, if we want to check whether a certain property $\Phi$ is satisfied by some graph in $\rho(L)$, then it is enough that we are able to check that locally. The key idea is to cut first $L$ into pieces using the algebraic property that a regular language $L$ has a finite syntactic monoid $M_L$. Hence,

---

1 We briefly discuss our encoding of graphs as words (and some related work) in Sect. 1.2.
Our result shows that, for every regular language \( L \), we obtain an important saturation property: Whenever \( w \in L \), we define the set of words \([w]\) to be all words in the same congruence class of \( w \). So, \( \rho([w]) \) plays the role of an open neighborhood around the graph \( \rho(w) \). Inside each \( \rho([w]) \), we define finitely many “smallest” graphs. Thereby, we find a finite set of finite graphs \( \mathcal{F} \) such that the collection of these finitely many graphs still has the entire information about \( \rho(L) \). In order to reveal that information, we construct for each \( F \in \mathcal{F} \) a (possibly infinite) graph \( F^\infty \). The graph \( F^\infty \) contains \( F \) as an induced subgraph; and \( F^\infty \) comes with a graph morphism onto the graph \( F \in \mathcal{F} \). The structure of that infinite graph is fully explicit and actually easy to understand. For example, it might happen that \( F \) consists of a single edge between two endpoints and \( F^\infty \) is the complete infinite bipartite graph \( (\mathbb{N}, \mathbb{N}, \mathbb{N} \times \mathbb{N}) \). Our result shows that, for every subset \( \mathcal{C} \) of \( L \), we have \( G \in \rho(L) \) if and only if for some \( F, G \in \rho(L) \) appears as a finite subgraph of \( F^\infty \) containing \( F \).

Our geometric approach to \( \rho(L) \) has two steps. First, we use the algebraic notion of syntactic monoid. The second step is a graph theoretical definition of \( F^\infty \). The outcome of following this road map is Corollary 3. It tells us that (with respect to our positive and negative decidable results) it is enough to consider only four different classes \( \mathcal{C}_1 \subset \cdots \subset \mathcal{C}_4 \) of graphs \( \rho(L) \).

1. \( \rho(L) \in \mathcal{C}_1 \) if and only if the set \( \rho(L) \) is finite.
2. \( \rho(L) \in \mathcal{C}_2 \) implies that \( \rho(L) \) has bounded tree-width.
3. \( \rho(L) \in \mathcal{C}_3 \) implies that every connected finite bipartite graph appears as a connected component of some \( G \in \rho(L) \).
4. \( \rho(L) \in \mathcal{C}_4 \) implies that every connected finite graph appears as a connected component of some \( G \in \rho(L) \).

Moreover, if \( L \) is regular, then we can compute the smallest \( \ell \) such that \( \rho(L) \in \mathcal{C}_1 \). Caveat: it may happen that \( \rho(L) \) is in \( \mathcal{C}_3 \) and in addition it contains arbitrarily large connected non-bipartite graphs, but nevertheless \( \rho(L) \notin \mathcal{C}_4 \).

Since the syntactic monoid of a regular language is finite, we find some \( t, p \in \mathbb{N} \) with \( p \geq 1 \), threshold and period, such that for every \( n \in \mathbb{N} \) there is some \( c \leq t + p - 1 \) with \( b^c \equiv_L b^p \) where \( \equiv_L \) denotes the syntactic equivalence. The pair \( (t, p) \) tells us that \( b^c \equiv_L b^p \) implies first, \( n = c \) for all \( 0 \leq c < t \) and second, \( b^p \equiv_L b^{p+1} \) if and only if \( n \geq t \). This is the key observation when proving that we have no more than these four classes above. If \( L \subseteq \mathcal{G} \) is not regular, then the syntactic monoid \( M_L \) is infinite. Still there are interesting examples where \( M_L \) satisfies the Burnside condition that all cyclic submonoids of \( M_L \) are finite. If so, then there exist \( t, p \in \mathbb{N} \) with \( p \geq 1 \) such that the syntactic properties stated above hold for the powers of the letter \( b \). In this case, we say that \( L \) satisfies the \((b, t, p)\)-torsion property.

Theorem 1 shows that for every subset \( L \subseteq \mathcal{G} \) satisfying the \((b, t, p)\)-torsion property, there exists a regular set \( R \subseteq \mathcal{G} \) such that \( \rho(L) = \rho(R) \). This is quite an amazing result. Its proof relies on the fact that \( \rho(L) \) is determined once we know the Parikh image of \( rf(L) \) in \( \mathbb{N}^{t+p-1} \), where for \( w \in \mathcal{G} \), the reduced form \( rf(w) \) is obtained by replacing every \( b^p \) by \( b^c \), where \( c \) is the smallest \( 0 \leq c \leq t + p - 1 \) such that \( b^c \equiv_L b^p \). Hence, for deciding whether some graph \( G \in \rho(L) \) satisfies a property, we can assume that language \( L \) is regular. We are interested in decidable properties \( \Phi \), only. Thus, we assume that the set \( \{G \mid G \models \Phi \} \subseteq \mathcal{G} \) is decidable. If \( \rho(L) \) is finite, then we can compute all graphs in \( \rho(L) \) and we can output all \( G \in \rho(L) \) satisfying \( \Phi \).

Finiteness of \( \rho(L) \) is actually quite interesting and important. It is a case where a representation of \( L \) by a DFA or a regular expression can be used for data compression. The minimal size of a regular expression (or the size of a DFA) for \( L \) is never worse than listing all graphs in \( \rho(L) \), but it might be exponentially better. For a concrete case, we refer to Exam-
ple 3. The compression rate becomes even better if we use a context-free grammar which produces a finite set \( L \) of words in \( \Sigma^* \), only. In the extreme case, \( L \) is a single word \( w \). Then, it might happen that the grammar (or straight-line program) is exponentially more succinct than writing \( w \) as a word in \( \Sigma^* \). Thus, possibly we can decide the existence of a graph in \( \rho(L) \) satisfying \( \Phi \) even though \( L \) is highly compressed by the chosen graph representation.

If \( L \) is regular, the existence of planar graphs in \( \rho(L) \) is conceptually easy to decide: Given \( L \), we can compute a number \( n(L) \in \mathbb{N} \) such that \( \rho(L) \) contains a planar graph if and only if there is some \( w \in L \) of length at most \( n(L) \) such that \( \rho(w) \) is planar. On a meta-level, whenever we were able to decide whether some \( G \in \rho(L) \) satisfies \( \Phi \), then we found effectively a corresponding number \( n(L) \). Moreover, positive decidability results are easy to establish for typical graph properties like “planarity” and many other graph properties.

Membership in the second class \( C_2 \) implies that \( \rho(L) \) has bounded tree-width. In this case, by [5, 6, 26] we know that when given any property \( \Phi \) which is definable in monadic second-order logic, MSO for short, then it is decidable whether there is a graph in \( \rho(L) \) satisfying \( \Phi \). Languages \( L \subseteq \mathbb{G} \) such that first, \( \rho(L) \) has finite tree-width, and second, the \((b, t, p)\)-torsion property holds, can be visualized as a set of graphs sharing some finite subgraph as a backbone structure to which arbitrarily large stars can be glued. This observation leads to Theorem 5: The satisfiability problem for MSO sentences is decidable for languages in the second class. Not surprisingly, the proof of Theorem 5 uses Theorem 1.

For the other two classes, the picture changes drastically: The first-order theory, FO for short, becomes undecidable [27]. Conversely, we are not aware of any “natural” graph property \( \Phi \) (which is not encoding Turing machine computations) where the satisfiability problem for \( \Phi \) is not trivial for \( C_3 \) and \( C_4 \). For example, for these classes \( \rho(L) \) contains non-planar graphs, because \( \rho(L) \) contains a graph where the complete bipartite graph \( K_{2,3} \) is a connected component. For a similar reason, \( \rho(L) \) contains graphs without any perfect matching. It is therefore more interesting to know whether some \( G \in \rho(L) \) allows a perfect matching. This problem is decidable, as we show, but the decision procedure is more involved.

1.2 Encoding of graphs and related work

Our encoding of graphs by using words over a binary alphabet is quite natural but obviously not unique. For instance, one could use larger alphabets, say, a unique letter per vertex, in writing down vertex or edge lists. As we use a code to write down vertex and edge names, we could interpret our encoding also as using a larger alphabet. However, in the context of the questions that we discuss in this paper, this would lead to automata over infinite alphabets, and we wanted to avoid discussing these here.

The bit complexity of \( \gamma(G) \) (encoding an edge-graph \( G \) with \( n \) vertices and \( m \) edges) is \( O(n \cdot m) \) and hence as good as traditional incidence matrices. More compact representations seem to lead to encodings that are not fit to be tested by finite automata and are hence avoided.

With the idea of using larger alphabets, still completely different encodings are possible. For instance, Kitaev and Seif introduced in [15] a representation of directed acyclic graphs by associating vertices to sets of letters of a word. This is also interesting for our discussions, as [15, Thm. 1.8]) yields a characterization of the Word Problem of Perkins’s semigroup \( \mathbf{B}_2^1 \) in terms of graph problems. In [3], again different interpretations of words as graphs and also typical graph problems are investigated for these encodings. Also, Bera and Mahalingam [3] draw connections to Parikh images.

In [17], Kuske generalizes results of de Malo and de Oliveira Oliveira in [7] on second-order finite automata by using automatic structures. As an application, Kuske shows in [17,
Thm. 3.6] how to decide typical properties of languages classes accepted by second-order finite automata.

Although our results go beyond regular sets $L$, the focus and the motivation come from a situation when $L$ is regular. A typical question could be whether there exists some planar graph in $\rho(L)$. Solving this type of decision problems was the motivation to study regular realizability problems in [1, 24, 28] and, independently, calling them $\text{int}_{\text{Reg}}$-problems\(^2\) in [12, 29–31].

## 2 Notation and preliminaries

Some of the following notation was introduced and explained in the introduction, Sect. 1. For convenience, we repeat them. We let $\mathbb{N} = \{0, 1, 2, \ldots\}$ be the set of natural numbers and $\mathbb{N}_\infty = \mathbb{N} \cup \{\infty\}$. Throughout, if $S$ is a set, then we identify a singleton set $\{x\} \subseteq S$ with the element $x \in S$. The power set of $S$ is identified with $2^S$ (via characteristic functions). If $E \subseteq X \times Y$ is a relation, then $E^{-1}$ denotes its inverse relation $E^{-1} = \{(y, x) \in Y \times X \mid (x, y) \in E\}$. By $\text{id}_X$, we mean the identity relation. Recall that $Y^X$ denotes the set of mappings from a set $X$ to a set $Y$. If $f : X \rightarrow Y$ and $g : Y \rightarrow Z$ are mappings, then $gf : X \rightarrow Z$ denotes the mapping defined by $gf(x) = g(f(x))$. If convenient, we abbreviate $f([x])$ as $f[x]$. As usual, a mapping $f : X \rightarrow Y$ can be also lifted to $f : 2^X \rightarrow 2^Y$.

Henceforth, $\Gamma$ denotes a finite alphabet, and $\Gamma^*$ denotes the free monoid over $\Gamma$. Any subset of $\Gamma^*$ is a language. If $S \subseteq \Gamma^*$ is a language, then, in our paper, $S^\infty$ has the same meaning as $S^\infty = \{a_1 \cdots a_k \in \Gamma^* \mid k \geq 1, a_1, \ldots, a_k \in S\}$.

Recall that a subset $C \subseteq \Gamma^*$ is a code if $c_1 \cdots c_m = d_1 \cdots d_n \in \Gamma^*$ with $c_i, d_j \in C$ implies that $m = n$ and $c_i = d_j$ for all $1 \leq i \leq m$. For a word $w = a_1 \cdots a_n \in \Gamma^*$ with $a_i \in \Gamma$, we let $\overline{w} = a_n \cdots a_1$ be the reversal of $w$. That is, we read the word $w$ from right to left.

Each alphabet is equipped with a linear order on its letters.\(^3\) The linear order on $\Gamma$ induces the short-lex linear order $\preceq_{\text{slex}}$ on $\Gamma^*$. That is, for $u, v \in \Gamma^*$, we let $u \preceq_{\text{slex}} v$ if either $|u| < |v|$ or $|u| = |v|, u = pcu'$, and $v = pdv'$ where $c, d \in \Gamma$ with $c < d$. Here, $|u|$ denotes the length of $u$. Similarly, $|u|_c$ counts the number of occurrences of letter $c$ in $u$. We also fix a binary alphabet $\Sigma = \{a, b\}$ with the order $a < b$.

### 2.1 Monoids

A monoid $M$ is a semigroup $(M, \cdot)$ with a neutral element $1 \in M$. If we use a multiplicative notation, then $1$ denotes the neutral element of a monoid. In particular, the empty word in free monoids is denoted by $1$ as well. In commutative monoids, we might use an additive operation, and then the neutral element is denoted as $0$. This is standard and there will be no risk of confusion. Cyclic monoids are commutative because, by definition, they are generated by a single element. Every finite cyclic monoid $M$ is defined by two numbers $t, p \in \mathbb{N}$ with $p \geq 1$ (where $t$ is the threshold and $p$ is the period) such that $M$ is isomorphic to the quotient monoid $C_{t, p}$ of $(\mathbb{N}, +, 0)$ with the defining relation $t = t + p$. Hence, the carrier set of $C_{t, p}$ equals $\{0, 1, \ldots, t + p - 1\}$. If $t = 0$ and $p = 1$, then $C_{t, p}$ is the trivial monoid $\{0\}$.

---

\(^2\) The notation $\text{int}_{\text{Reg}}$ refers to intersection non-emptiness with regular languages.

\(^3\) This convention is standardized in DIN 44300 and ISO 2382.
If $M$ is a monoid, then $u \leq v$ means in our paper $v \in MuM$. That is, $u$ is a factor of $v$. This notation applies, in particular, to the monoids $\Gamma^*$ and $N^\Gamma$. Here, $N^\Gamma$ denotes the free commutative monoid over $\Gamma$. Since $\Gamma$ is finite, $N^\Gamma$ is the set of mappings from $\Gamma$ to $\mathbb{N}$. Its elements are called vectors.

2.1.1 Syntactic monoids, congruences, and the word problem

Every subset $L \subseteq \Gamma^*$ has a syntactic monoid $M = M_L$, see for example [10]. The elements of $M_L$ are the congruence classes $[u] = \{ v \in \Sigma^* | v \equiv_L u \}$ w.r.t. the syntactic congruence $\equiv_L$ which is defined by the following equivalence.

$$u \equiv_L u' \text{ if and only if } \forall x, y \in \Sigma^*: xuv \in L \iff xu'v \in L$$

If $L$ is regular, then $M_L$ is finite. Later, we do not need that $M_L$ is finite, so we will relax this condition. It will suffice that the letter $b$ appears in $\Gamma$ and is generating a finite submonoid.\(^4\)

Let $\varphi: \Gamma^* \to G$ be a surjective homomorphism onto a finitely generated group $G$. Then, the Word Problem of $G$ denotes the set $\text{WP}(G) = \{ w \in \Gamma^* | \varphi(w) = 1 \}$. If this set is decidable, then we say that Word Problem of $G$ is decidable, because on input $u, v \in \Gamma^*$, we can decide whether $\varphi(u) = \varphi(v)$. It is a classical fact (and an easy exercise) that decidability of Word Problem does not depend on the generating set and that the syntactic monoid of $\text{WP}(G)$ is the group $G$ itself; we refer to [2].

2.1.2 Burnside groups

Recall that $|\Sigma| = 2$. The free Burnside group $B(2, p)$ is defined as the quotient

$$B(2, p) = \Sigma^*/\{ x^p = 1 \mid x \in \Sigma^* \}$$

where $p \geq 1$. It is a group, because every $x$ has the inverse element $x^{p-1}$ thanks to $p \geq 1$.

For $p$ large enough, Adjan has shown in the 1970s that $B(2, p)$ is infinite, answering a question of Burnside dating back in its original form to 1902. Actually, Adjan also showed the decidability of the Word Problem of $B(2, p)$ if $p$ is large enough. Here, a group (with two generators) is called $p$-periodic if it is the homomorphic image of some $B(2, p)$. Kharlampovich constructed in [14] a periodic group $B(2, p)$ with a generating set $\Sigma$ and a finite set of words $w_1, \ldots, w_r \in \Sigma^*$ such that the group $B(2, p)$ has the monoid presentation $B(2, p) = \Sigma^*/\{ w_i = 1, w^p = 1 \mid 1 \leq i \leq r, w \in \Gamma^* \}$ (as an abstract group) and where the Word Problem $\text{WP}(B(2, p))$ is undecidable. Thus, the language $L = \text{WP}(B(2, p))$ is undecidable, nevertheless $w^p \equiv_{\Sigma^*} 1$ for all $w \in \Sigma^*$. We use this example to illustrate that there are undecidable languages satisfying the $b$-torsion property which will be defined in Sect. 3.

2.2 Parikh-images

If $v, w \in \Gamma^*$, then $|w|_v$ denotes the number how often $v$ appears as a factor in $w$, i.e.,

$$|w|_v = |\{ u \in \Gamma^* \mid \exists s : uvw = w \}|.$$ 

If $P \subseteq \Gamma^*$, then the Parikh-mapping w.r.t. $P$ is defined by $\pi_P : \Gamma^* \to \mathbb{N}^P$, mapping a word $w \in \Gamma^*$ to its Parikh-vector $(|w|_v)_{v \in P} \in \mathbb{N}^P$.

\(^4\) According to Sect. 3, we will call this the $b$-torsion property.
The traditional case is $P = \Gamma$; then the Parikh-vector becomes $(|w|_a)_{a \in \Gamma}$ and the Parikh-mapping is the canonical homomorphism from the free monoid $\Gamma^*$ to the free commutative monoid $\mathbb{N}^\Gamma$. As usual, $\mathbb{N}^\Gamma$ is partially ordered such that

$$u \leq v \iff \forall z \in \Gamma : u(z) \leq v(z).$$

Subsets $L \subseteq \mathbb{N}^\Gamma$ which can be written as $L = q + \sum_{i \in I} \mathbb{N}p_i$, with $I$ finite, are called linear and a finite union of linear sets is called semi-linear. We rely on the following classical results.

**Proposition 1**

1. **The complement of a semi-linear set $L \subseteq \mathbb{N}^\Gamma$ is effectively semi-linear.**

   Hence, the family on semi-linear sets is an effective Boolean algebra, see [11].

2. **The Parikh image of a context-free language is effectively semi-linear, see [22].**

A subset $S \subseteq \mathbb{N}^\Gamma$ is called **positively downward-closed** if first $v(z) \geq 1$ for all $v \in S$, $z \in \Gamma$ and second, $u \leq v \in S$ and $u(z) \geq 1$ for all $z \in \Gamma$ imply $u \in S$. The complement of a positively downward-closed set $S \subseteq \mathbb{N}^\Gamma$ is upward-closed, i.e., $u \geq v \in S$ implies $u \in S$. An upward-closed set $S$ is determined by its set $\min(S)$ of minimal elements. Dickson’s lemma says that the set $\min(S)$ is finite for all $S \subseteq \mathbb{N}^\Gamma$. Hence, every upward-closed subset is semi-linear. It follows by Proposition 1 (1) that every positively downward-closed set $S \subseteq \mathbb{N}^\Gamma$ is semi-linear, too. This observation is crucial for proving Theorem 1.

### 2.3 Graphs

All graphs are assumed to be (at most) countable, given as a pair $G = (V, E)$ where $E \subseteq V \times V$. An **undirected graph** is the special case where $E = E^{-1}$, so that $E$ describes the adjacency relation. If $G = (V, E)$ is a directed graph, then $G$ also defines the **undirected graph** $(V, E \cup E^{-1})$; and it defines the **undirected graph without self-loops** $(V, (E \cup E^{-1}) \setminus \text{id}_V)$. A graph without isolated vertices is called an **edge-graph**. Hence, the set of edges determines an edge-graph. If $G' = (V', E')$ and $G = (V, E)$ are graphs such that $V' \subseteq V$ and $E' \subseteq E$, then $G'$ is a **subgraph** of graph $G$ and we denote this fact by $G' \leq G$. If $U \subseteq V$ is any subset, then $G[U] = (U, E \cap U \times U)$ denotes the induced subgraph of $U$ in $G$. A subset $U$ is called independent if $G[U]$ is without any edge. A graph morphism $\varphi : (V', E') \to (V, E)$ is given by a mapping $\varphi : V' \to V$ such that $(u, v) \in E'$ implies $(\varphi(u), \varphi(v)) \in E$. If $(V', E')$ and $(V, E)$ are undirected graphs without self-loops, then $\varphi : (V', E') \to (V, E)$ is a graph morphism as soon as $(\varphi(u), \varphi(v)) \in E \cup \text{id}_V$. We say that $\varphi$ is a projection if $\varphi$ is surjective on vertices and edges, i.e., $\varphi(V') = V$ and $\varphi(E') = E$. We consider graphs up to isomorphism, only. Hence, writing $G = G'$ means that graphs $G$ and $G'$ are isomorphic. According to the following Sect. 2.4, a graph $F = (V, E)$ is a retract of a graph $F' = (V', E')$ if there are morphisms $\varphi : F' \to F$ and $\gamma : F \to F'$ such that $\varphi \gamma$ is the identity on vertices and edges of $(V, E)$. Hence, $F$ appears in $F'$ as the induced subgraph $F'[\gamma(V)]$. Another way to say this is that $F$ is an induced subgraph of $F'$ and there is a morphism $\varphi : F' \to F$ which is the identity on $F$.

We consider several special graphs (and graph properties) in our paper. By a **star**, we denote a graph $(V, E)$ such that there exists a vertex $z \in V$ with the property $E = \{(z, s) \mid s \in V \setminus \{z\}\}$. Thus, a star has a center $z$ and the directed edges are the outgoing rays of the star. We also use this notion to refer to an undirected connected graph where all but possibly one vertex have degree one. Let $G = (V, E)$ be an undirected graph. The graph $G$ is called a **clique** if all possible edges (apart from self-loops) are contained in $E$. The graph $G$ is called **bipartite** if its vertex set $V$ can be partitioned into $V_1$ and $V_2$ such
that there are no edges between vertices of the same class of the partition. A bipartite graph is called complete if no further edges can be added without violating bipartiteness. By \( K_n \), we denote a clique with \( n \) vertices, and \( K_{n,m} \) denotes the complete bipartite graph with \( n \) vertices in one of the classes and \( m \) vertices in the other one. By \( C_n \), we denote a cycle with \( n \) vertices. A graph is called acyclic if it does not contain any \( C_n \) with \( n \geq 3 \) as a subgraph. A nonempty, acyclic, and connected graph is called a tree. A vertex set is independent if its vertices are pairwise non-adjacent. A vertex set \( C \) is a vertex cover if for each edge, at least one endpoint belongs to \( C \).

Some of our results revolve tree-width. The notation is due to Robertson and Seymour, see [23]. It is one of the cornerstones in their famous graph minor project.\(^5\) Let \( G = (V, E) \) be a graph. A tree decomposition is specified by a tree \( T = (V_T, E_T) \) together with a mapping \( t : V_T \to 2^V \) such that the following properties are satisfied:

\[- \bigcup_{x \in V_T} t(x) = V; \]

\[- \text{ for each } (u, v) \in E, \text{ there is at least one } x \in V_T \text{ such that } \{u, v\} \subseteq t(x); \]

\[- \text{ for each } v \in V, \text{ the subgraph induced by } t^{-1}(B(v)) \text{ in } T, \text{ where } B(v) = \{b \in t(V_T) \mid v \in b\}, \text{ is connected.} \]

Elements in \( t(V_T) \) are also called bags. The greatest cardinality of any bag of a tree decomposition is called the bag-size of the tree decomposition, and its tree-width is defined by the bag-size minus one. The tree-width of a graph is the smallest tree-width of any tree decomposition of this graph. In the special case when the tree of a tree decomposition is a path, we also speak of a path decomposition. The path-width of a graph is the smallest tree-width of any path decomposition of this graph. For example, the tree-width of a graph \( G \) is at most one if and only if \( G \) is acyclic, but the path-width of acyclic graphs cannot be bounded.

In our paper, every word \( w \in \Sigma^* \) represents a directed finite graph \( \rho(w) = (V(w), E(w)) \) together with a linear order on vertices as follows:

\[ V(w) = \{ \text{abi} a \mid \text{ibi} a \leq w \} \]

\[ E(w) = \{ (\text{abi} a, \text{bji} a) \mid \text{abi} a \times \text{bji} a \leq w \} \]

The empty word represents the empty graph: there are no vertices and no edges. We extend \( \rho \) to \( 2^{\Sigma^*} \) by \( \rho(L) = \{ \rho(w) \mid w \in L \} \). Vice versa, if \( G = (V, E) \) denotes a finite graph with a linear order on its vertices, then, for \( 1 \leq i, j \in \mathbb{N} \), the \( i \)th vertex is represented by the factor \( \text{abi} a \), and an edge from the \( i \)th vertex to the \( j \)th vertex is represented by the factor \( \text{abi} a \text{aibi} a \). Thus, vertices are encoded by elements in the set \( \mathbb{V} = \{ \text{abi} a \mid 1 \leq i \in \mathbb{N} \} \) and edges are encoded by elements in the set \( \mathbb{E} = \{ \text{abi} a\text{aibi} a \mid 1 \leq i, j \in \mathbb{N} \} \). Note that \( \mathbb{V} \cup \mathbb{E} \) is an infinite regular code. Using these conventions, the regular set \( \Sigma^* \setminus \mathbb{G} \) represents the empty graph, as well as its subset \( \Sigma^* \setminus \mathbb{G} \) represents all finite graphs. The same property holds for the complement \( \Sigma^* \setminus \mathbb{G} \): It represents all finite graphs, too. Indeed, \( \mathbb{G} \cap a\mathbb{G} = \emptyset \) but \( \rho(\mathbb{G}) = \rho(\mathbb{aG}) \), since the words \( w \in \mathbb{G} \) and \( aw \not\in \mathbb{G} \) share the same set of factors from \( \mathbb{V} \cup \mathbb{E} \). In contrast to \( \mathbb{G} \), infinitely many nonempty words in \( \Sigma^* \setminus \mathbb{G} \) represent the empty graph, for example all words without any \( b \) or with at most one \( a \). The set \( \mathbb{E}^* \) represents all edge-graphs, i.e., all graphs without isolated vertices. Every nonempty finite graph has infinitely many representations in \( \mathbb{G} \). For example, there are uncountably many subsets \( L \subseteq (\text{abi} a)^+ \subseteq \mathbb{V}^+ \) and each \( \rho(L) \) represents nothing but the one-point graph without self-loop. In order to choose a unique (and minimal) representation for a finite graph \( G = (V, E) \), we choose the minimal word

\(^5\) According to Wikipedia (April 2022) the concept of tree-width is due to Bertelè and Brioschi [4]. Later it was rediscovered by Halin (1976) [13], before Robertson and Seymour came up with it in [23]. According to [9], Robertson and Seymour were apparently unaware of the earlier papers.
\(\gamma(G) = u_1 \cdots u_m v_1 \cdots v_n \in \mathcal{G}\) in the short-lex ordering on \(\Sigma^*\) such that \(\rho \gamma(G) = G\), \(u_k \in \mathbb{E}\) for \(1 \leq k \leq m\) and \(v_\ell \in \mathcal{V}\) for \(1 \leq \ell \leq n\). Each \(u_k\) is of the form \(ab^i aab^j a\) representing an edge and each \(v_\ell\) is of the form \(ab^l a\) representing an isolated vertex. We call \(\gamma(G)\) the short-lex representation of \(G\). Since \(\gamma(G)\) is minimal w.r.t. \(\leq_{\text{slex}}\), we have \(m = |E|\) and \(n\) is the number of isolated vertices. For a graph without isolated vertices, i.e., an edge-graph, this means that it is given by its edge-list. The set of all \(\gamma \rho(\mathcal{G})\) is context-sensitive but not context-free. The \(\text{uvwxy-Theorem}\) (i.e., the context-free pumping lemma) does not hold for \(\gamma \rho(\mathcal{G})\). For instance, all edge-graphs in \(\gamma \rho(\mathcal{G})\) with \(n\) vertices must be represented by vertex names \(ab^i a\) with \(1 \leq i \leq n\), admitting no ‘holes’ in this vertex name interval, as otherwise there would be a smaller short-lex representation of some graph. Such a property is not maintained by pumping.

\[2.4\text{ Retractions and retracts}\]

Let \(\rho : X \to Y\) and \(\gamma : Y \to X\) be mappings between sets. (This holds more general for mappings which are morphisms in some category.) If \(\rho(\gamma(y)) = y\) for all \(y \in Y\), then \(\rho\) is called a retraction and \(Y\) is called a retract of \(X\) with section \(\gamma\). We also say that \(\rho^{-1}(y)\) is the fiber of \(y \in Y\). For example, if \(\rho : X \to Y\) is a homomorphism of groups \(X\) and \(Y\) and \(H = \ker(\rho)\) is the kernel, then \(\rho\) is a retraction if and only if \(X\) is a semi-direct product of \(H\) by \(Y\). Another example comes from formal languages: let \(X\) be the set of deterministic finite automata (DFAs) where every state is reachable from the start state. Then, the well-known minimization process defines a retraction to the set of minimal DFAs.

Later in Sect. 4.5, we define a marked graph as a triple \((V_F, E_F, \mu)\), where \((V_F, E_F)\) is a finite graph and \(\mu \subseteq V_F \cup E_F\) is the set of marked vertices and edges. Then, \((V_F, E_F, \mu) \mapsto (V_F, E_F)\) defines a retraction by letting \(\gamma(V_F, E_F) = (V_F, E_F, \emptyset)\).

Let \(\mathcal{G}\) be the set of finite graphs, \(\rho : \mathcal{G} \to \mathcal{G}\) be the interpretation of words as graphs according to our encoding, and \(\gamma : \mathcal{G} \to \mathcal{G}\) be the encoding of a graph by its short-lex normal form. Then, \(\rho\) is a retraction. Retractions are a main tool to understand \(\rho(L)\) if \(L\) is regular or more general, if \(L\) satisfies the \(b\)-torsion property, as defined in the next section.

3 The \(b\)-torsion property

We are interested in properties of graphs which are specified by languages \(L \subseteq \mathcal{G}\). If \(L\) can be arbitrary, then we can specify uncountably many families of graphs. So, we cannot expect any interesting and general positive (decidability) results. As a minimal request, we restrict our attention to subsets \(L \subseteq \mathcal{G}\) where membership for \(\rho(L)\) is decidable.\(^6\) As a matter of fact, membership for \(\rho(L)\) might be decidable although membership for \(L\) is undecidable. As we will see in Corollary 1, the following definition yields a sufficient condition that membership for \(\rho(L)\) becomes decidable.

Definition 1 Let \(\Gamma\) be a finite alphabet containing the letter \(b\).

\(-\) A language \(L \subseteq \Gamma^*\) satisfies the \((b, t, p)\)-torsion property for \(t, p \in \mathbb{N}\) with \(p \geq 1\) if we have: \(b^t \equiv_L b^{t+p}\).

\(-\) A language \(L \subseteq \Gamma^*\) satisfies the \(b\)-torsion property if there are \(t, p \in \mathbb{N}\) with \(p \geq 1\) such that \(L\) satisfies the \((b, t, p)\)-torsion property.

\(^6\) Notice that it does not really matter for this question how the input graph is presented; for simplicity, we can assume graph \(G\) to be tested for membership is given by a word \(w\) such that \(\rho(w) = G\).
Every regular language $L \subseteq \Gamma^*$ satisfies the $b$-torsion property because the syntactic monoid $M_L$ is finite. The $b$-torsion property is a strong restriction if $L$ is not regular. For example, $K = \{ a^n b^n \mid n \in \mathbb{N} \}$ does not have that property since $b^k \equiv_L b^m \iff k = m$. The context-free language $L = \{ w a w \mid w \in \{ a b a b^2 a \}^+ \}$ is not regular, but it satisfies the $b$-torsion property for $t = 3$ and $p = 1$. In this case, $\rho(L)$ is a not very interesting set of a few small graphs. The next example shows that there are (non-regular) context-free languages satisfying a $(b, t, p)$-torsion property where $\rho(L)$ is infinite.

**Example 1** Let $C$ be a nonempty finite alphabet and $K \subseteq C^*$ be context-free. Let $h : C^* \to \mathbb{E}^*$ be a homomorphism. That is, $h$ is defined by words $h(c) \in (ab^+aaba^b+a)^*$ for $c \in C$. Even if $h(K)$ is infinite, the set $\rho(h(K))$ is still finite. Indeed, let $t = \max \{ i \in \mathbb{N} \mid \exists c \in C : ab^i a \leq h(c) \}$, then graphs in $\rho(h(K))$ have at most $t$ vertices. Let us make the context-free language $h(K)$ larger by closing $h(K)$ under rewriting rules $b \to b^{1+p}$. Context-free languages are closed under adding context-free rewriting rules. We obtain a new context-free language $L$ with $h(K) \subseteq L \subseteq \mathbb{E}^*$ and where $L$ satisfies the $(b, t, p)$-torsion property. We claim that $\rho(L)$ is a very rich and an infinite family of graphs (in contrast to the finite set $\rho(h(K))$).

We content ourselves to consider the case $p = 1$. For $p = 1$, it is rather easy to see that every nonempty finite edge-graph appears in $\rho(L)$: we have $\rho(L) = \rho(\mathbb{E}^*)$. Let $G = (V_G, E_G) \in \rho(\mathbb{E}^*)$ and $m = |E_G|$ be the number of edges. Since $h(K)$ is infinite, there is some edge $e = ab^i aaba^b a$ with $1 \leq i, j \leq t$ such that $e$ appears in some $w_f \in L$ at least $m$ times as a factor. Then, thanks to $p = 1$, we have $w = (ab^i aaba^b a)^f \in L$, where $m \leq \ell$ and $\ell$ can be chosen as large as we wish. The graph $\rho(w)$ is a one-point graph with a self-loop. Let us come back to the graph $G$. Without restriction, we have $V_G \subseteq \{ t, \ldots, \ell \}$ by shifting the names of vertices and therefore their encoding. Now, for each $(u, v) \in E_G$, one after the other, we replace one factor $ab^i aaba^b a$ in $w$ by the factor $ab^k aaba^b a$. This changes the word $w$, but the new word $w$ still belongs to $L$, because we shifted the names of vertices beyond the threshold $t$ and we have $p = 1$. By creating, if necessary, several copies of the same edge, the procedure yields a word $w_G$ such that $w_G \in L$ and $\rho(w_G) = G$. □

As soon as all cyclic submonoids of $M_L$ are finite, $L$ satisfies the $b$-torsion property for all letters $b \in \Gamma$. For example, consider the Word Problem of any free Burnside group $B(2, p)$. All of them satisfy the $b$-torsion property. Almost all of the groups $B(2, p)$ are infinite and therefore the corresponding Word Problems are not regular. If it is not regular, then the Word Problem of $B(2, p)$ is even not context-free by [21], since a periodic group cannot have any non-trivial free group of finite index.

For the rest of the paper, if $L \subseteq \Sigma^*$ satisfies the $b$-torsion property, then $t, p \in \mathbb{N}$, standing for threshold and period, denote those natural numbers such that the cyclic submonoid generated by the letter $b$ in the syntactic monoid $M_L$ is isomorphic to $C_{t, p}$. That is, we have $t, p \in \mathbb{N}$ with $p \geq 1$, where $t + p$ is minimal such that

$$\{ [b^n] \mid n \in \mathbb{N} \} = \{ [b^c] \mid 0 \leq c \leq t + p - 1 \}.$$  \hspace{1cm} (1)

Moreover, we assume that $L$ is specified such that on input $n \in \mathbb{N}$, we can compute the value $0 \leq c \leq t + p - 1$ with $b^n \equiv_L b^c$. This assumption is satisfied if $L$ is regular and specified, say, by some NFA. For $L \subseteq \mathbb{G}$, we have $[ab^c a] = a[b^c]a$ and $[ab^c aaba^b a] = a[b^c]a a a a [b^d] a$. The tacit assumption is important for the next definition to compute, for example, the reduced form according to the next definition.

**Definition 2** Let $L \subseteq \mathbb{G}$ satisfy the $(b, t, p)$-torsion property. For every $[b^n]$, we define its reduced form by $\text{rf}([b^n]) = [b^c]$ if $[b^c] = [b^n]$ and $0 \leq c \leq t + p - 1$. Given $w \in \mathbb{G}$,
we define the reduced form \( rf(w) \) by replacing every factor \( ab^m a \leq w \) by \( a \rf[b^m]a \). The saturation \( \hat{w} \) of \( w \) is defined by replacing every factor \( ab^m a \leq w \) by the set \( a[b^m]a \). Hence, \( rf(w) \in \hat{w} \subseteq \mathbb{G} \).

**Remark 1** Let \( L \subseteq \mathbb{G} \) satisfy the \((b, t, p)\)-torsion property. By possibly decreasing the numbers \( t \) and/or \( p \), we may assume that for every \( 1 \leq c \leq t + p - 1 \), there is some \( w \in L \) such that \( ab^c a \leq rf(w) \). Moreover, we have \([b^c] = \{b^c\}\) if and only if \( c < t \).

**Lemma 1** Let \( L \subseteq \mathbb{G} \) satisfy the \((b, t, p)\)-torsion property. Then, for every \( w \in \mathbb{G} \),

\[
\begin{align*}
  w \in L & \iff \hat{w} \subseteq L \iff rf(w) \in L .
\end{align*}
\]

**Proof** Trivial, by definition of the \((b, t, p)\)-torsion property.

### 4.1 Examples

The following examples serve as an introduction to a more general situation we will face later.

**Example 2** In the following, we let \( R \subseteq \mathbb{G} \) and \( t, p \in \mathbb{N} \) with \( p \geq 1 \) such that \( b^p \equiv_R b^{n+p} \) for all \( n \geq t \), i.e., \( R \) satisfies the \((b, t, p)\)-torsion property. Moreover, we let \( 1 \leq c < t \) such that \([b^c] = \{b^c\}\).

1. Let \( w \in R \subseteq (ab^c aaab^p (b^p)^* a)^+ \subseteq \mathbb{G} \) for a fixed \( n \in \mathbb{N} \) with \( t \leq n < t + p \). This implies that \( w \) contains a factor \( ab^c aaab^d a \) with \( t \leq d < t + p \) and \([b^d] = b^{d+p} \). We have \( w \in (ab^c aaab^p (b^p)^* a)^m \) for \( m = |w|_a/5 \). Hence, \( w = (ab^c aaab^{d_1} a) \cdots (ab^c aaab^{d_m} a) \) where \( d_i = n + k_i p \) with \( k_i \in \mathbb{N} \) for \( 1 \leq i \leq m \). The set \( \{ d_i \mid 1 \leq i \leq m \} \) can have any cardinality \( s \) in \( \{1, \ldots, m\} \). Therefore, \( \rho(w) \) is a single star with at least one ray and at most \( m \) rays. If \( R \) is finite, then \( \mathcal{F} = \rho(R) \) is an effective finite collection of stars with at least one ray and at most \( r \) rays where \( r = \max \{|w|_a/5 \mid w \in R \} \).
Recall that $M_R$ denotes the syntactic monoid of $R$. We claim that $\mathcal{F}$ is infinite if and only if there is some $M \geq |M_R|$ such that $(ab^c aaab^{p+a})^M \in R$. Moreover, if $\mathcal{F}$ is infinite, then $\mathcal{F}$ is the set of all finite stars with at least one ray. The claim holds if $\sup \{|w|/5 \mid w \in R\} < \infty$, as in this case $\mathcal{F}$ is finite. Thus, let $\sup \{|w|/5 \mid w \in R\} = \infty$. Then, there is some $w \in R$ such that $ab^c aaab^{p+a}$ appears at least $|M_R|$-times as a factor. This implies that there is some $M \geq |M_R|$ such that $(ab^c aaab^{p+a})^M \in R$. The claim follows.

One can show that $S = (abaaab^2 b^* a)^*(aba)$ is locally testable and therefore star-free. Hence, the set of all finite stars is specified by a star-free subset of $\Sigma^*$.

2. If $R \subseteq (ab^c aaab^d b^+ a)^* ab^e a \subseteq \mathbb{G}$, then $\rho(R)$ is set of stars with center $ab^c a$ and outgoing rays to vertices $ab^d a$ where $d > c$. Moreover, the following dichotomy holds: The set of stars in $\rho(R)$ is either finite or it contains almost all finite stars. Indeed, $\rho(R)$ is a set of stars with center $c$, possibly without rays. If $\rho(R)$ is finite we are done. Otherwise, let $\rho(R)$ be infinite. Then, for each $r \in \mathbb{N}$, there is a star in $\rho(R)$ with more than $r$ rays. This implies that for all $r \in \mathbb{N}$ there is some $\ell \geq r$ and a word $w \in R$ which has more than $\ell$ pairwise different factors $ab^c aaab^{p+k} a$ with $r \leq d_i$. If $r$ is large enough, then each of these factors can be replaced by a factor $ab^c aaab^{c_i} a$ where we have that $t < c_i \leq t + p$. This yields a word $w' \in R$. Now, $w'$ is very long as $\ell$ is very large. Hence, we can factorize the word $w' = uw''v$ such that first, $w''$ contains one of these factors $ab^c aaab^{c_i} a$ and second, $S = u(w''^* + v \subseteq R$. Since $ab^c aaab^{c_i} a \equiv_R ab^c aaab^{c_i} + pk a$ for all $k \in \mathbb{N}$, we conclude that $\rho(S)$ contains almost all stars, i.e., all stars but finitely many that are missing. 

The $(b, t, p)$-torsion property is trivially satisfied if $L \subseteq \mathbb{G}$ is a finite set. An interesting case motivated by data compression. As mentioned in Sect. 1: if $L$ is finite, then the minimal size of a regular expression for $L$ is never worse than listing all graphs in $\rho(L)$, but it might be exponentially better, as the following example shows. This type of data compression is important and well known. It is used in DNA computing and bio-inspired modeling, frequently. It is the basis of practical algorithms like RePair, too. See, e.g., [18, 19].

**Example 3** Let $G = (V, E)$ be a connected planar graph with vertex set $V = \{1, \ldots, n\}$ with $n \geq 3$ together with an embedding into the two-dimensional sphere. For every subset $S \subseteq \{n + 1, \ldots, 2n\}$, let $G_S$ denote the graph. $G_S = (V \cup S, E \cup \{(s, s - n) \mid s \in S\})$. The
family \( C_n = \{ G_S \mid S \subseteq \{n + 1, \ldots, 2n\} \} \) might contain exponentially many graphs in \( n \). This happens, for example, if \( G \) is a cycle of \( n \) nodes. Then, \( C_n \) has more than \( 2^n/2n \in 2^{O(n)} \) connected planar graphs. If we embed \( G \) in the two-dimensional sphere where the additional edges are spikes pointing out of the sphere, then \( G_S \) can be visualized as a discrete model of a three-dimensional “crown with at most \( n \) cusps.” A two-dimensional representation of a full crown (having all possible cusps) is depicted in Fig. 1a, while Fig. 1b shows the situation when some cusps were chosen to be removed. It is straightforward to write down a \( 2 \)\( n \)-fold concatenation of finite sets which describes a finite set \( L_n \subseteq \mathbb{G} \) such that \( \rho(L_n) = C_n \). The size of the corresponding regular expression is \( O(n^2) \). Thus, we have a polynomial-size blueprint potentially producing a family of exponentially many mutations of a single “corona”, the Latin word for “crown.”

4.2 Introducing new alphabets

After seeing a couple of examples, we introduce certain subsets of \( \mathbb{G} \) as alphabets to express subsets of graphs. This prepares the geometric viewpoint to view a graph as a point in the \( d \)-dimensional space \( \mathbb{N}^d \) if the size of the chosen alphabet is \( d \). If \( L \) is regular, then the dimension depends on \( L \) and it can be quite large, but it is computable by using the reduced form of words in \( L \). Below, all this will be explained in detail.

Let \( \ell \in \mathbb{N} \). Depending on \( \ell \), we define two finite and disjoint sets that we consider as alphabets:

\[
A_\ell = \{ ab^m a a b^n a \mid 1 \leq m, n \leq \ell \} \subseteq \mathbb{E},
\]

\[
B_\ell = \{ ab^m a \mid 1 \leq m \leq \ell \} \subseteq \mathbb{V}.
\]

Note that \( A_\ell = B_\ell a B_\ell \). By \( C_\ell \), we denote the union of \( A_\ell \) and \( B_\ell \), which is also a finite alphabet with a linear order between letters given by the following definition:

\[
x \preceq_\ell y \iff xy \in A_\ell B_\ell \lor (xy \in (A_\ell A_\ell \cup B_\ell B_\ell) \land x \preceq_{\text{slex}} y).
\]

We have \( A_\ell \cap B_\ell = \emptyset \) and \( \mathbb{N}^C \ell \) has dimension \( d = \ell(\ell + 1) \). Actually, \( C_\ell \) is a code, i.e., \( c_1 \cdots c_m = d_1 \cdots d_n \in \Sigma^* \) with \( c_i, d_j \in C_\ell \) implies \( m = n \) and \( c_i = d_i \) for all \( 1 \leq i \leq m \). Hence, \( \preceq_\ell \) is well defined by Eq. (4).

**Definition 3** Let \( L \subseteq \mathbb{G} \) satisfy the \( b \)-torsion property. Then, we define an alphabet \( C_L \subseteq C_\ell \) where \( \ell \) is large enough such that for every \( w \in L \) its reduced form \( rf(w) \) can be written as a word in \( C_L^* \). To make the definition unique, we choose \( C_L \) to be the smallest set \( C_L \subseteq \bigcup_{\ell \in \mathbb{N}} C_\ell \) which has this property.

If the context to \( L \) is clear, then we drop the index \( L \) and we denote by \( C \) any subset of some \( C_\ell \) such that \( C_L \subseteq C \subseteq C_\ell \). This flexibility is useful if we wish to introduce new vertices or new edges and we need “fresh” names in \( C \) for them. In such a situation, we also write \( A, B \) (without subscript \( \ell \)). In order to distinguish the factor ordering and the linear ordering defined by Eq. (4), we denote the latter by \( \preceq_C \).

The linear order \( \preceq_C \) on \( C \) defines a corresponding short-lex ordering on \( \Sigma^* \). Moreover, if \( uvv \in C^+ \) with \( x \in A \) and \( u, v \in \Sigma^* \), then \( u, v \in C^* \). The analogue for \( y \in B \) does not hold, in general. For example, \( aba \in B \) and \( abaaaba \in C \), but \( aaba \notin C \). As \( C \) is a code, the inclusion \( C \subseteq \Sigma^* \) yields an embedding \( h_C : C^+ \to \Sigma^* \). Let \( w \in C^* \) and \( G = \rho(w) \). If \( L = \rho^{-1}(G) \), then the minimal element in \( h_C^{-1}(L) \) with respect to the short-lex ordering for words in \( C^* \) is a word in \( A^* B^* \). It is the same as the minimal element in \( h_C(h_C^{-1}(L)) \). Springer
with respect to the ordering $a < b$. For example, both $w_1 = abbaabbaaabaabbaaabaabbaaabaabba$ and $w_2 = ababbaabbaaabaabba$ represent the same graph with vertex names $aba$, $abba$, $abbba$, and a single edge, but the word $w_2$ comes before $w_1$ in the short-lex ordering for words in $C^*$. Yet, there is still a smaller presentation of a graph with three vertices and a single edge, consider namely $w_3 = abaaabbaabbaabba$.

### 4.3 The power of $b$-torsion

Recall that for any subset $P \subseteq \Sigma^*$ and $v \in \Sigma^*$ we denote by $\pi_P(v) \in \mathbb{N}^P$ its Parikh image as defined in Sect. 2.2. The following lemma shows a crucial “downward closure property” used in Theorem 1.

**Lemma 2** Let $L$, $C$, and $rf$ as in Definition 2 and in Definition 3. Let $v \in C^*$ and $w \in L$ such that $\pi_C(v) \leq \pi_C(rf(w))$. If $\pi_{\{z\}}(v) \geq 1$ for all $z \in C$, then we have $\rho(v) \in \rho(L)$.

**Proof** If $\pi_{\{z\}}(v) = \pi_{\{z\}}(rf(w))$ for all $z \in C$, then $\rho(v) = \rho(w)$ and therefore $\rho(v) \in \rho(L)$ because $\rho(w) \in \rho(L)$ thanks to $w \in L$. Thus, we may assume that $1 \leq \pi_{\{z\}}(v) < \pi_{\{z\}}(rf(w))$ for some $z \in C$. Thus, without restriction we have $rf(w) = uzu'zu''$ with $z \in C$ and $u, u', u'' \in C^*$ and $\pi_C(v) \leq \pi_C(rf(w'))$ where $w' = uu'zu''$. We have $rf(w') = w'$ by Definition 2. Moreover, a repetition of some $z \in C$ does not change the specified graph. Let $\tilde{w}'$ denote the saturation of $w'$. By Lemma 1, we have $\rho(\tilde{w}') \subseteq \rho(L)$. Define $L' = L \cup \tilde{w}'$. Then, we have $w' \in L'$ and $\rho(L) = \rho(L')$; moreover, $L'$ satisfies the same $(b, t, p)$-torsion property as $L$ does. We can work with the same $C$, too. Since $w'$ is shorter than $w$, we conclude, by induction on the length of $w$, that $\rho(v) \in \rho(L') = \rho(L)$.

**Theorem 1** Let $L \subseteq \mathbb{G}$ be any language satisfying the $b$-torsion property. Then, there is a regular set $R \subseteq \mathbb{G}$ such that $\rho(L) = \rho(R)$.

**Proof** The proof relies on Dickson’s lemma. We have $\rho(L) = \bigcup \{\rho(\tilde{v}) \mid \tilde{v} \in L\}$ by Lemma 1. The set $\rho(rf(L))$ is finite, as $L$ satisfies the $b$-torsion property. Thus, there is a finite subset $K \subseteq L$ such that $\rho(rf(L)) = \bigcup \{\rho(v) \mid v \in K\}$. Let $C \subseteq E \cup V$ be the finite subset such that

$$C = \left\{z \in E \cup V \mid \exists v \in K : \pi_{E \cup V}(v(z)) \geq 1 \right\} .$$

Splitting $L$ into disjoint subsets and replacing $C$ by various subsets of $C$ we may assume without restriction that for all $w \in L$, the corresponding $C$ contains (after the split) exactly all letters that are factors of $rf(w)$ and moreover, for all $v \in C^*$, we have $z \leq rf(w) \iff \pi_C(w(z)) \geq 1$. After this modification, there are no vectors in $\pi_C(rf(L))$ with zero entries. The crucial observation is stated in Lemma 2. The lemma tells us that we do not change $\rho(L)$ if we augment $L$ by all words $v \in C^*$ where there is some $w \in L$ such that $1 \leq \pi_C(w(z)) \leq \pi_C(rf(w))(z)$ for all $z \in C$. Therefore, we may assume without restriction that $\pi_C(rf(L))$ is positively downward-closed according to Sect. 2.2. We also explained in Sect. 2.2 that Dickson’s lemma implies that $\pi_C(rf(L))$ is semi-linear. By Parikh’s theorem, see Proposition 1 (1), there is a regular set $R' \subseteq C^*$ of words such that $\pi_C(R') = \pi_C(rf(L))$. The class of regular sets is closed under regular substitutions. The inclusion $C \subseteq (V \cup E)^*$ defines a canonical homomorphism $h : C^* \to (V \cup E)^*$. Hence, if we substitute in $R'$ every letter $z$ by $[h(z)]$, then we obtain $\rho(R) = \rho(L)$, where $R = \bigcup \{\tilde{w} \mid w \in R'\}$.

**Corollary 1** Let $L \subseteq \mathbb{G}$ satisfy the $b$-torsion property. Then, given a finite graph $G = (V_G, E_G)$ as an input, it is decidable whether $G \in \rho(L)$.

The contents of Corollary 1 inspired the title of this subsection.
Proof By Theorem 1, we may replace \( L \) by some regular set \( R \) where \( \rho(R) = \rho(L) \). In particular, we can calculate the threshold \( t \) and the period \( p \) such that \( R \) satisfies the \((b, t, p)\)-torsion property. The set \( \rho(\tau_f(R)) \) is finite and effectively computable. For every \( F \in \rho(\tau_f(R)) \), we compute the short-lex normal form \( \gamma(F) \) as defined above. We obtain a finite set \( W \) of words containing all those \( \gamma(F) \). Thus, the set \( \hat{W} = \bigcup \{ \hat{w} \mid w \in W \} \) is effectively regular and it holds that \( \rho(\hat{W}) = \rho(R) \). Let \( m = \max \{|w| \mid w \in W \} \). Let \( M = mt|V_G||E_G| \). Hence, \( G \in \rho(R) \) if and only if there is a word \( w \in \hat{W} \) of length at most \( M \) such that \( G = \rho(w) \).

Corollary 2 Let \( L \subseteq G \) be context-free satisfying the \((b, t, p)\)-torsion property. Then, we can effectively calculate a regular set \( R \subseteq G \) such that \( \rho(R) = \rho(L) \).

Proof Let \( C = \{ z \in \mathbb{V} \cup \mathbb{E} \mid \forall m \in \mathbb{N} : ab^m a \leq z \Rightarrow m \leq t + p - 1 \} \). The inclusion of \( C \subseteq (\mathbb{V} \cup \mathbb{E})^* \) defines a homomorphism \( h \) from the free monoid \( C^* \) to \( (\mathbb{V} \cup \mathbb{E})^* \). Hence, \( L' = h^{-1}(L) \) is effectively context-free. Therefore, \( \pi_C(L') = \pi_C(h^{-1}(L)) \) is effectively semi-linear. This semi-linear set can be represented by a regular language \( R' \subseteq C^* \). As in the proof of Theorem 1, we obtain \( R \) as the image of \( R' \) under the regular substitution which replaces every letter \( z \in C \) by \( [h(z)] \).

Let \( R \subseteq G \) be regular. It is well known that there might be a much more concise representation by some context-free language \( K \subseteq G \) such that \( \pi_C(K) = \pi_C(R) \) and hence \( \rho(K) = \rho(R) \). Therefore, we might describe graph families even more concisely using context-free grammars than using NFAs.

### 4.4 Switching the alphabet and Parikh images

By Theorem 1, we know that regular languages suffice to describe all sets \( \rho(L) \) where \( L \subseteq G \) satisfies the \( b \)-torsion property. Therefore, we restrict ourselves to regular languages. Throughout this section, \( R \subseteq G \) denotes a regular language. Hence, we can calculate a threshold \( t \) and a period \( p \geq 1 \) such that \( R \) satisfies the \((b, t, p)\)-torsion property. Since \( R \) is regular, the set \( L = h^{-1}(R) \cap A^*B^* \) is regular, too; its Parikh image \( \pi_C(L) \subseteq \mathbb{N}^C \) is effectively semi-linear.\(^7\) Hence, for some finite index sets \( J \) and \( I_j \) we can write

\[
\pi_C(L) = \bigcup_{j \in J} \left( q_j + \sum_{i \in I_j} \mathbb{N}p_i \right),
\]

where \( q_j, p_i \in \mathbb{N}^C \) are vectors. Splitting \( \pi_C(L) \) into more linear sets by making the index set \( J \) larger and the sets \( I_j \) smaller (if necessary), we can assume without restriction that, for all \( j \in J \) and \( z \in C \), we have \( \sum_{i \in I_j} p_i(z) \leq q_j(z) \). To see this, let \( 1 \in I_j \). Then, we have

\[
q_j + \sum_{i \in I_j} \mathbb{N}p_i = \left( q_j + \sum_{i \in I_j \setminus \{1\}} \mathbb{N}p_i \right) \cup \left( q_j + p_1 + \sum_{i \in I_j} \mathbb{N}p_i \right).
\]

Splitting \( L \) into even more but finitely many cases, we can assume without restriction (for simplifying the notation) that the set \( J \) is a singleton. Thus, \( \pi(L) = q + \sum_{i \in \mathbb{I}} \mathbb{N}p_i \) for some \( q, p_i \in \mathbb{N}^C \) such that \( \sum_{i \in \mathbb{I}} p_i(z) \leq q(z) \) for all \( z \in C \). Moreover, making \( A, B, C \) perhaps

\(^7\) The same assertion, including the effectiveness, holds, by Parikh’s theorem, if \( R \) is context-free. Therefore, our results still hold if \( R \) is context-free language with a known \((b, t, p)\)-torsion property.
smaller, we may assume without restriction that \( q(z) \geq 1 \) for all \( z \in C \) and \( C = A \cup B \).

(A similar argument was used in the proof of Theorem 1.)

In order to understand the set of graphs in \( \rho(R) \), it suffices to understand the set of finite graphs defined by linear sets of the form \( S = q + \sum_{i \in I} \mathbb{N} p_i \subseteq \mathbb{N}^C \), where \( q(z) \geq 1 \) for all \( z \in C \) and \( \sum_{i \in I} p_i \leq q \). For that purpose, we let \( r = \sum_{i \in I} p_i \leq q \) and we define a function \( \alpha : C \rightarrow \mathbb{N}_\infty \) as follows.

\[
\alpha(z) = \begin{cases} 
q(z) \text{ if } r(z) = 0 \land \exists m \in \mathbb{N} : t \leq m \land ab^m a \leq z \
\infty \text{ if } r(z) \geq 1 \land \exists m \in \mathbb{N} : t \leq m \land ab^m a \leq z \
1 \text{ otherwise. That is: } \forall m \in \mathbb{N} : ab^m a \leq z \implies m < t.
\end{cases}
\]

For all \( z \in C \), let \( L_z \subseteq \Sigma^* \). Then, we introduce the following notation:

\[
\prod_{z \in C} L_z = L_{z_1} \cdots L_{z_{|C|}}
\]

(7)

where \( z_i \leq z_j \) for all \( i \leq j \) according to the linear order defined in Eq. (4). Observe that \( \prod_{z \in C} L_z \) is regular if all \( L_z \) are regular. With this notation, we define regular sets \( W_\alpha \subseteq \Sigma^* \) and \( L_\alpha \subseteq \Sigma^* \) by

\[
W_\alpha = \prod_{z \in C} z^{\alpha(z)} \quad \text{and} \quad L_\alpha = \prod_{z \in C} [z]^{\alpha(z)}
\]

(8)

Here, as said in Sect. 2, the notation \([z]^{\infty}\) means \([z]^+\). Note that \( \rho(W_\alpha) \) is a single graph defined by the word \( w_\alpha = \prod_{z \in C} z \). Indeed, \( \rho(\prod_{z \in C} z^{e_z}) \) does not depend on the exponents \( e_z \) as long as \( e_z \geq 1 \) for all \( z \in C \).

**Theorem 2** The sets \( W_\alpha \) and \( L_\alpha \) are regular sets with \( W_\alpha \subseteq L_\alpha \) and \( \rho(L_\alpha) = \rho(R) \).

**Proof** Without restriction, Eq. (5) reads as \( \pi_C(L) = q + \sum_{i \in I} \mathbb{N} p_i \). As \( z \in [z] \), the inclusions \( W_\alpha \subseteq L_\alpha \) and \( \rho(W_\alpha) \subseteq \rho(R) \) follow by definition. For the converse, let \( v \in R \) and \( G = \rho(v) \). Choose some \( w \in L_\alpha \) with \( \pi_C(w) = q + \sum_{i \in I} m_i p_i \). Choosing \( m \in \mathbb{N} \) large enough, we find \( \pi_C(v) \leq q + mr \in \pi_C(\rho(L_\alpha)) \) where, as above, \( r = \sum_{i \in I} p_i \). Hence, we can apply Lemma 2 to finish the argument. \( \square \)

### 4.5 Marked graphs

In the next steps, we define for \( \alpha \) a finite family of finite graphs \( \mathcal{F}_\alpha \). Adding a marking to the graphs in \( \mathcal{F}_\alpha \) leads to a finite family \( \mathcal{F}_\beta \) of marked graphs, denoted by \((V_i, E_i, \mu_i)\), where \( \mu_i \subseteq V_i \cup E_i \) reflects the marking on vertices and edges. Then, for each \( F_i = (V_i, E_i, \mu_i) \), we define a possibly infinite graph \( F_i^\infty \). Knowing all \( F_i^\infty \), we will recover \( \rho(R) \) as retracts. We are now defining the abstract notion of a marked graph, where some vertices and edges are marked.

**Definition 4** A marked graph is a tuple \( F = (V_F, E_F, \mu) \), where \((V_F, E_F)\) is a finite graph and \( \mu \subseteq V_F \cup E_F \) denotes the set of marked vertices and edges. Isolated vertices may appear, but if an isolated vertex is marked, then there is exactly one isolated vertex. We also require that whenever an edge \((u, v)\) is marked, then at least one of its endpoints is marked, too. A marked edge-graph is a marked graph without isolated vertices.

In order to construct a family of marked graphs, we begin with the definition of a function \( \alpha' \) (depending on \( \alpha \)) which in turn yields \( \mathcal{F}_\alpha' \).
Definition 5 For $z \in C$ let $\alpha'(z) = \alpha(z)$ if $\alpha(z) < \infty$ and $\alpha'(z) = 1$, otherwise. We define $F_a = \rho(L_{\alpha'})$, where $L_{\alpha'} = \prod_{z \in C} [\alpha'(z)]$ is analogous to Eq. (8).

Since $\alpha'(z)$ is never equal to $\infty$, the set $\rho(L_{\alpha'})$ is a finite set of finite graphs. Indeed, if a graph belongs to $\rho(L_{\alpha'})$, then the number of edges is upper-bounded by $\sum_{z \in A} \alpha'(z)$ and the number of isolated vertices is upper-bounded by $\sum_{z \in B} \alpha'(z)$. Clearly, there are only finitely many graphs of that form and we can compute them all. We use a more concrete representation by annotating vertices and edges. An annotation of a graph $(V, E)$ is a mapping $\nu$ from $V \cup E$ to some set $S$. If $S = \{0, 1\}$, then we speak about a marking and $\nu$ is defined through the subset $\mu = \nu^{-1}(1)$ of marked elements. In our case, we annotate a vertex (or an edge, respectively) $f$ by $\nu(f) = z$ in $B$ (or $A$, respectively). It follows $\rho\left(\prod_{e \in E} \nu(e) \cdot \prod_{v \in V} \nu(v)\right) = G$. The number how often an annotation $z \in C$ can appear is upper-bounded by $\alpha'(z)$.

Therefore, there is a finite set of annotated graphs

$$L = \{(V_1, E_1, \nu_1), \ldots, (V_k, E_k, \nu_k)\} \quad \text{(with } k = |L|)$$

such that it holds that $F_a = L$ when forgetting the annotation.\footnote{Annotated graphs are equal if there is a graph isomorphism between the underlying graphs which respects the annotation.} For the verification whether $(V, E, \nu) \in L$ we can use words in $A^*B^*$ which are of bounded length and a guess-and-check algorithm. For each $(V, E) \in F_a$, we guess its annotation $\nu$. Next, we check that the guess is correct as follows. Each vertex $u \in V$ belongs to some class $a[b^c]a$ such that $\nu(u) = a[b^c]a$. The idea is to give for the vertex $u$ a unique name of the form $ab^{c+m_u}p$. For that, we scan through all vertices in any order. Let $u$ belong to $a[b^c]a$. If $c \leq t$, then we choose $m_u = 0$: the name of $u$ becomes $ab^{c}a$. In the other case, if $c \geq t$, then we choose the smallest $m_u$ such that the name $c + m_u p$ is still available. Thus, the number of concrete names for vertices is identical to the number of vertices, and the annotation is now coded directly in the names. This process yields concrete names for the edges, too. Finally, we verify that our construction is a concrete realization of $(V, E)$.

We now turn $L$ into a family $F_\beta$ of marked graphs $\{(V_1, E_1, \mu_1), \ldots, (V_k, E_k, \mu_k)\}$. Thus for every $1 \leq i \leq k$, the set of marked graphs contains some $(V_i, E_i, \mu_i)$. However, it may happen that $(V_i, E_i, \nu_i) \neq (V_j, E_j, \nu_j)$ as annotated graphs, but $(V_i, E_i, \mu_i) = (V_j, E_j, \mu_j)$ as marked graphs. Thus, it may happen that $|F_\beta| < k$. See Example 4 where even $|F_a| < |F_\beta| < k$.

To this end and in general, we mark each graph $G$ according to $\alpha'(z)$. We mark the isolated vertices of $G$ as follows. If there is an isolated vertex of the form $ab^{c+m_u}pa$ with $m_u \geq 1$, then we mark exactly one isolated vertex and we leave all other isolated vertices unmarked. Let $I$ be the set of isolated vertices of $G$. Then, $G$ belongs to the set of graphs in $\rho(R)$ where either the number of isolated vertices is exactly $|I|$ if no isolated vertex is marked and where there are at least $|I|$ isolated vertices in the other case.

Next, we consider the set of edges. An edge has the form $e = ab^{c+m_u}paab^{d+m_v}pa$ such that $e \in [z]$ for some $z = ab^{c}aabb^{d}a \in A$. We mark the edge $e$ if and only if $\alpha(z) = \infty$ (which implies $[z] \neq \{z\}$). In the case that $e$ becomes marked, we know that $m_u + m_v \geq 1$. If $m_u \geq 1$, then we also mark the endpoint $ab^{c+m_u}pa$. If $m_v \geq 1$, then we also mark the endpoint $ab^{d+m_v}pa$.

The construction shows that by making, if necessary, the alphabet $C = A \cup B$ as well as the threshold $t$ (perhaps much) larger, but by keeping the period $p$, we assume that for each $1 \leq i \leq k$, there is some $\beta_i : C \rightarrow \mathbb{N}_\infty$ such that first, $\rho(\prod_{z \in C} z) = (V_i, E_i) \in L$ and second,
where \( \beta_i(z) \in \{0, 1, \infty\} \) with the requirements that first, \( \beta_i(z) = \infty \implies [z] \neq \{z\} \) and second, if \( z = uav \) denotes an edge with \([z] \neq \{z\}\), then \( \beta_i(z) = \infty \iff (\{u\} \neq \{u\} \vee \{v\} \neq \{v\}) \). Finally, we can shrink \( C \) again such that actually, \( \beta_i(z) \in \{1, \infty\} \) for all \( z \in C \). Moreover, we can ensure that we have \( \beta_i(z) = \infty \) for at most one \( z \in B \) designating isolated vertices. The advantage of \( \beta_i \) is that it gives a direct encoding of the underlying marked graph.

**Example 4**  Consider the following regular language.

\[
W = (ab^1aaab^5(bb)^*a)(ab^5(bb)^*aaab^3a)(ab^3aaab^2a)
\]

\[
(ab^2aaab^4(bb)^*a)(ab^4(bb)^*aaab^1a)^+\]

Its syntactic monoid satisfies the \((b, t, p)\)-torsion property with threshold \( t = 4 \) and period \( p = 2 \). Hence, \([b^k] = [b^k] \) for \( k = 1, 2, 3 \), while \([b^4] \) and \([b^5] \) are infinite: \([b^4] = \{b^k \mid k \geq 4 \land k \text{ is even}\} \), and likewise, \([b^5] = \{b^k \mid k \geq 5 \land k \text{ is odd}\} \). We have \( W \subseteq \mathbb{E}^+ \) and it defines an infinite set of edge-graphs thanks to the final factor \((ab^k(bb)^*aaab^1)^+\).

Indeed, for all \( d \geq 2 \), there is a graph in \( \rho(W) \) where the vertex \( ab^1a \) has degree \( d \).

For better readability, let us abbreviate an edge \( ab^1aaab^5a \) in the standard “edge notation” as \((i, j)\). Since we adjust the alphabet \( C = A \) in Eq. (8) such that \( \alpha(z) \geq 1 \) for all \( z \in C \), we obtain

\[
C = \{(1, 5), (5, 3), (3, 2), (2, 4), (4, 1)\}.
\]

According to Definition 5, we have \( \alpha'(z) = 1 \) and \( \alpha(z) = \infty \iff z = (4, 1) \) for all \( z \in C \).

The family \( \mathcal{F}_\alpha \) is a singleton. We have \( \mathcal{F}_\alpha = \{\rho(w)\} \) where \( w \) can be chosen to be the word

\[
w = (ab^1aaab^5a)(ab^2aaab^4a)(ab^3aaab^2a)(ab^4aaab^1a)(ab^5aaab^3a) \in C^5.
\]

The resulting (undirected) graph \( \rho(w) \) is a cycle on five vertices, known as a \( C_5 \). As vertex set we may choose \( \{1, \ldots, 5\} \) and the edge set is given by \( C \). We obtain a single marked graph where the edge \((4, 1)\) is marked and hence the vertex 4. However, the family \( \mathcal{F}_\beta \) is larger.

For instance, it contains two marked graphs which are both marked paths with six vertices, each of them being a marked \( P_6 \), but they are not isomorphic as marked graphs; see Fig. 2. However, on the level of \( \mathcal{F}_\alpha \), we forget the marking, and then both words describe the same finite graph, which is a \( P_6 \), i.e., a path on six vertices. Actually, Fig. 2 shows concrete names for vertices. But names for 4 and 6 (or 5 and 7, respectively) can be interchanged. Thus, a set of annotated graphs is larger than two.

By Theorem 2, we have \( \rho(L_\alpha) = \rho(R) \). Since \( \rho(R) \) is *cet obscur objet du désir*, we perform another splitting according to Eq. (9). For understanding \( \rho(R) \), it is enough to understand the family of graphs defined by each \( \beta_i \), or, what is the same, to understand the set of graphs defined by each marked graph in \( \mathcal{F}_\beta \).

Actually, in many cases we do not have to compute all marked graphs. It is sufficient if a subset \( K \subseteq \{1, \ldots, k\} \) satisfies \( \bigcup_{i \in K} \rho\left(\prod_{z \in C} [z]^{\beta_i(z)}\right) = \rho(L_\alpha) \). For example, if we are interested in deciding whether there is a graph in \( \rho(L_\alpha) \) that satisfies a property \( \Phi \) and if \( \Phi \) is a non-trivial property of undirected bipartite graphs which is true if \( \Phi \) holds in some connected component of size, say, \( n - 1/n \) for a graph with \( n \) vertices, then finding a single \( z \in B \) with \( \alpha(z) = \infty \) is enough to say “YES.”

---

9 Luis Buñuel Portolés, 1977.
Let us switch to a more abstract viewpoint. We let $\mathcal{F}$ be any finite family of marked graphs. For each $F = (V_F, E_F, \mu) \in \mathcal{F}$, we define a possibly infinite graph $F^\infty$ where $(V_F, E_F)$ appears as an induced subgraph, and we define a family $\mathcal{G}_F$ of finite graphs. In our application, we consider finitely many $\mathcal{F}_\beta$, and then we study $\bigcup \{ \mathcal{G}_F | F \in \mathcal{F}_\beta \}$, where $F = (V_F, E_F, \mu)$ is the marked graph obtained by the canonical marking procedure described above (which might have removed isolated vertices). It turns out that, for a full description of $\rho(R)$, it is enough to describe sets $\mathcal{G}_F$ for marked graphs $F = (V_F, E_F, \mu)$. This requires to define $F^\infty$.

**Definition 6** Let $F = (V_F, E_F, \mu)$ be a marked graph as in Definition 4. Then, the graph $F^\infty = (V_F^\infty, E_F^\infty)$ is defined as follows.

$$V_F^\infty = V_F \times \{0\} \cup \bigcup_{u \in V_F} \{(u, k) | u \text{ is marked } \land k \in \mathbb{N}\},$$

$$E_F^\infty = E_F \times \{0\} \cup \{(k, k), (v, \ell) \in V_F^\infty \times V_F^\infty | (u, v) \in E_F \land (u, v) \text{ is marked}\},$$

with $E_F \times \{0\} = \{(u, 0), (0, v) | (u, v) \in E_F\}$. The family $\mathcal{G}_F$ is the set of finite subgraphs of $F^\infty$ containing $(V_F \times \{0\}, E_F \times \{0\})$ as an induced subgraph.

Observe that $F^\infty = (V_F, E_F)$ if and only if there is no marking, i.e., if $\mu = \emptyset$. We embed $F$ into $F^\infty$ by a graph morphism $\gamma$ which maps each vertex $u \in V_F$ to the pair $\gamma(u) = (u, 0) \in V_F^\infty$. The projection onto the first component $\varphi(u, k) = u$ yields a retraction for every $G \in \mathcal{G}_F$ with retract $F$. If no isolated vertex is marked, then $F^\infty$ has at most $|V_F|$ isolated vertices, but if there are marked vertices, then for every sufficiently large $k$, there is some graph in $\mathcal{G}_F$ which has exactly $k$ isolated vertices. In order to understand the graphs in $\mathcal{G}_F$ (which is our goal), it is enough to understand the graphs $G$ satisfying $F \leq G \leq F^\infty$. For $F = F^\infty$, we have the full information about that set. Thus, we focus on $F \neq F^\infty$.

Theorem 3 shows that $\rho(R)$ is rather rich as soon as some $F \in \mathcal{F}_\beta$ satisfies $F \neq F^\infty$.

The reader might find it helpful to look at the examples of Fig. 3 to understand the building of the graph $F^\infty$ from a marked graph $F$ if at least one vertex is marked. The pictures show five different situations: (a) a graph $F$ with a single isolated marked vertex: it gives rise to an arbitrary number of isolated vertices in $F^\infty$; (b) a graph $F$ with a single marked vertex incident to a marked edge yields an infinite star in $F^\infty$; (c) a graph $F$ with a marked edge with different endpoints where both are marked. It yields an infinite complete bipartite graph in $F^\infty$; (d) a graph $F$ with a marked self-loop gives rise to an infinite clique in $F^\infty$; (e) The

\[ \text{Springer} \]
rightmost figure combines the last three situations in a single marked graph, plus the effect of an unmarked edge.

**Theorem 3** Let \( F = (V_F, E_F, \mu) \) be any marked graph.

1. If there is no marking, then \( G_F = \{ (V_F, E_F) \} \).
2. If \( F \) contains a marked edge \((u, v)\) where the endpoint \( v \) is marked, then every finite star with center \((u, 0)\) appears as an induced subgraph of some \( G \in G_F \). Moreover, if we aim for a star \( S \) with \( r \) rays, then we find a graph \( G \) in \( G_F \) with at most \(|V_F| + r\) vertices that contains \( S \) as an induced subgraph.
3. Suppose we represent a bipartite graph as a triple \((U, V, E)\) where \( U \cap V = \emptyset \) and \( E \subseteq U \times V \). Let \( H \) be any finite bipartite edge-graph. If \( F \) contains a marked edge \((u, v)\) where \( u \) and \( v \) are marked, then a disjoint union of \( F \) and \( H \) appears in \( G_F \).
4. Let \( H \) be any finite graph. If \( F \) contains a marked self-loop \((u, u)\), then the disjoint union of \( F \) and \( H \) belongs to \( G_F \).
5. Let \( F \) be any marked graph such that at most two vertices are marked. Then, the following holds. A disjoint union of \( F \) and a triangle (or any other non-bipartite graph) appears in \( G_F \) if and only if there is some marked self-loop in \( F \).

**Proof** Define the vertex sets \( V_0 = V_F \times \{0\} \) and \( V_{\geq 1} = \{ (u, k) \mid u \in V_F \land k \geq 1 \} \). We consider the five cases separately.

1. By definition.
2. In the graph \( F^\infty \), there are directed edges from \((u, 0)\) to all \((v, k)\) with \( k \in \mathbb{N} \). Let \( S \) be a star with \( r \) rays. Consider

\[
V = V_F \cup \{ (v, k) \mid 1 \leq k \leq r \} \quad \text{and} \quad E = E_F \cup \{ (u, 0), (v, k) \mid 1 \leq k \leq r \}.
\]

Then, \((V, E) \in G_F \) and \( S \) appears as an induced subgraph of \((V, E)\) with center \((u, 0)\) as required.
3. and 4. Consider a marked edge \((u, v)\) with both endpoints marked. In the graph \( F^\infty \), there are directed edges from \((u, k)\) to all \((v, \ell)\), where \( k, \ell \in \mathbb{N} \). Consider the induced subgraph \( F^\infty[U] \) where \( U = V_{\geq 1} \). By definition, \( F^\infty[U] \) is disjoint from the subgraph \( F = F^\infty[V_0] \). For \( u = v \), the graph \( F^\infty[U] \) is an infinite complete graph; for \( u \neq v \), the graph \( F^\infty[U] \) is an infinite complete bipartite graph. The claims follow.
5. If \( F \) contains a marked self-loop, then, by definition, it is a self-loop around a marked vertex. Hence, we are done since every disjoint union of \( F \) and any other finite graph \( G \) appears in \( G_F \). For the other direction, assume that \( F \) has no marked self-loop. If a disjoint union of \( F \) and a finite non-bipartite graph \( G \) appears in \( G_F \), then we need at least three marked vertices to produce \( G \).

The following lemma uses the notions of vertex cover, of tree-width and of path-width, as defined above.

**Lemma 3** Let \( F = (V_F, E_F, \mu) \) be a nonempty marked graph such that each marked edge has at most one unmarked vertex incident to it. Then, for every \( G \in G_F \) both, the size of a minimum vertex cover and its path-width, are upper-bounded by \( |V_F| \), while its tree-width is upper-bounded by \( |V_F| - 1 \).

The following proof is based on well-known and standard techniques.

**Proof** Let \( G = (V_G, E_G) \in G_F \). Since every marked edge has at most one marked vertex incident to it, \( V_G \) is the disjoint union of \( V_F \) and an independent set \( U \). Hence, \( V_F \) is a vertex cover of \( G \). We now construct a tree decomposition as follows. We begin with a single bag \( B \) defined by the vertex cover \( V_F \). Then, for every \( u \in U \), we define a bag \( B_u \) by \( N(u) \cup \{u\} \) where \( N(u) \) is the set of neighbors of \( u \). Note that \( N(u) \subseteq V_F \). A bag \( B_u \) is always connected to \( B \). This is a tree decomposition of \( G \) which is a star with \(|U|\) rays. Alternatively, we can consider the bags \( B'_u = V_F \cup \{u\} \) and connect them all arbitrarily as a path.

For an illustration of our construction, we refer to Fig. 4.

**Remark 2** The construction in the proof of Lemma 3 is optimal with respect to the minimum vertex cover and to the tree-width if \( (V_F, E_F) \) is a clique and \( S = (V_F, S_F) \) is its subgraph of marked edges such that \( S \) is a star, where the center of the star is not marked. In general, we might achieve smaller vertex covers and tree-widths by beginning with a tree decomposition of \( F \).
Definition 7 For \( i = 1, \ldots, 4 \), we define classes \( C_i \) containing the sets \( \rho(L) \) where \( L \subseteq \mathcal{G} \) is regular. If \( \mathcal{G} \) denotes such a set \( \rho(L) \), then:

1. We let \( \mathcal{G} \in C_1 \) if \( \mathcal{G} \) is a finite set of graphs.
2. We let \( \mathcal{G} \in C_2 \) if the set \( \mathcal{G} \) has bounded tree-width.
3. We let \( \mathcal{G} \in C_3 \) if \( \mathcal{G} \in C_2 \) or if there exists a finite set of graphs \( \mathcal{F}' \) such that, for every finite bipartite graph \( G' \), there is some \( F' \in \mathcal{F}' \) such that the disjoint union of \( F' \) and \( G' \) appears in \( \mathcal{G} \).
4. We let \( \mathcal{G} \in C_4 \) if \( \mathcal{G} \in C_3 \) or if there exists a finite set of graphs \( \mathcal{F}' \) such that, for every finite graph \( G' \), there is some \( F' \in \mathcal{F}' \) such that the disjoint union of \( F' \) and \( G' \) appears in \( \mathcal{G} \).

Note that our definition enforces \( C_i \subseteq C_j \) for \( 1 \leq i \leq j \leq 4 \). The reason for introducing these classes is driven by our motivation to find graphs in the sets \( \rho(L) \) satisfying some property \( \Phi \). For that, let us consider a marked graph \( F = (V_F, E_F, \mu) \). Recall that \( \mu \) is a subset of \( V_F \cup E_F \). Now, if \( \mu' \subseteq \mu \), then \( \mathcal{G}_{F'} \subseteq \mathcal{G}_F \) where \( F' = (V_F, E_F, \mu') \). Thus, increasing the marking increases \( \mathcal{G}_F \), and this makes it more likely to find a graph satisfying \( \Phi \).

Corollary 3 Let \( F = (V_F, E_F, \mu) \) be a marked graph. Then, the following holds.

1. We have \( \mathcal{G}_F \in C_1 \) if and only if there is no marked vertex.
2. If there is no marked edge \((u, v)\) where both \( u \) and \( v \) are marked, then \( \mathcal{G}_F \in C_2 \). This implies that \( \mathcal{G}_F \) has bounded tree-width. The tree-width and the minimal size of a vertex cover are bounded by \( |V_F| \).
3. If there is a marked edge \((u, v)\) where \( u \) and \( v \) are marked, then \( \mathcal{G}_F \in C_3 \). This implies that every connected finite bipartite graph appears as a connected component of some \( G \in \mathcal{G}_F \). In particular, for every \( k \in \mathbb{N} \), there is some \( G \in \mathcal{G}_F \) whose tree-width and whose minimal size of a vertex cover are both greater than \( k \).
4. We have \( \mathcal{G}_F \in C_4 \) if and only if every connected finite graph appears as a connected component of some \( G \in \mathcal{G}_F \).

Proof It is enough to prove the lemma when \( \rho(L) = \mathcal{G}_F \) where \( F = (V_F, E_F, \mu) \) is a marked graph. The result is now a direct consequence of Theorem 3 and Lemma 3 for the second item concerning the bounded tree-width. \( \square \)

We now show in Corollary 4 that for star-free languages, the classification only contains three cases. As an example, consider \( R = (ab^+aaab^+a)^+ \); then \( R \) is star-free\(^{10} \) and \( \rho(R) \in C_4 \), because \( \rho(R) = \mathcal{G}_F \), where \( F \) is a marked self-loop around a marked vertex. Therefore, if \( \rho(L) \) equals \( \rho(R) \) for a star-free regular language \( R \), then Corollary 4 states that \( \rho(L) \in C_3 \) implies \( \rho(L) \in C_4 \). Hence, if \( R \) is star-free, then \( \rho(R) \) belongs to three classes, only.\(^{11} \)

\(^{10}\) The minimal DFA accepting \( R \) is easily seen to be counter-free in the sense of [20]. Alternatively, one can check with an FO sentence that a word \( w \) contains the factor \( aaa \), but no factor \( bab \). Every occurrence of the factor \( aaa \) of \( w \) is preceded by \( ab^+ \) and succeeded by \( b^+a \). Moreover, the word \( w \) has a prefix in \( ab^+aaa \) and a suffix in \( aaab^+a \) and if there is a factor \( bab \) in \( w \), then this factor is (immediately) preceded and followed by the factor \( aaa \). Finally, between any two occurrences of the factor \( aaa \) in \( w \), there must be a factor \( bab \).

\(^{11}\) Corollary 4 corrects a misprint in [8, Cor. 3].
Corollary 4 Let $R$ be a star-free language such that $\rho(R)$ is infinite. Then, we have either $\rho(R)\in\mathcal{C}_2$ or $\rho(R)\in\mathcal{C}_4 \setminus \mathcal{C}_2$.

Proof Recall Schützenberger’s classical theorem that a language $R$ is star-free if and only if its syntactic monoid $M_R$ is aperiodic, see [25]. Therefore, $R$ satisfies the $(b, t, p)$-torsion property for some threshold $t \geq 0$ with period $p = 1$. We may assume that $\rho(R)$ is infinite. Then, we find a marked graph $F$ such that $\mathcal{G}_F \subseteq \rho(R)$ and $\mathcal{G}_F$ is infinite. Consider $\mathcal{G}_F \notin \mathcal{C}_2$. We have to show that $\mathcal{G}_F \notin \mathcal{C}_3$. Hence, $F$ contains a marked edge where at least one endpoint is marked. This marked vertex is defined by the word $aab^t+1a$. Thus, it is unique because every vertex in $F$ has the form $ab^i a$ for $1 \leq i \leq t + 1$ and the vertices $aba, \ldots, ab^t a$ are not marked. Since $\mathcal{G}_F \notin \mathcal{C}_2$, the marked graph $F$ contains a marked self-loop $aab^t+1aaba^{t+1}a$. We now apply Theorem 3 and Corollary 3. A marked self-loop implies $\rho(R) \in \mathcal{C}_4 \setminus \mathcal{C}_3$. □

5 Graph properties

A graph property is a decidable subset $\Phi \subseteq \mathbb{G}$. For a finite graph $G$, we write $G \models \Phi$ if the short-lex representation $\gamma(G)$ belongs to $\Phi$.\footnote{The construction of the marked graphs changed perhaps the threshold, but not the period which is therefore still 1. For simplicity, the new threshold is still called $t$.} Given a word $w \in \mathbb{G}$, we can compute $\gamma(w)$. Hence, we can assume without restriction that $\Phi$ is saturated: $\rho^{-1}(\rho(\Phi)) = \Phi$. To simplify our presentation, we focus on properties of undirected finite graphs (without self-loops). This can be achieved by making the set $\Phi$ larger such that $\Phi$ has the following desired property: If $u \in \mathbb{G}$ represents the graph $\rho(u) = (V, E)$ and $\Phi$ speaks about undirected graphs (resp. undirected graphs without self-loops) then $\rho(u) \models \Phi \iff (V, E \cup E^{-1}) \models \Phi$ (resp. $\rho(u) \models \Phi \iff (V, (E \cup E^{-1}) \setminus \text{id}_V) \models \Phi$).

We are interested in the following (uniform) satisfiability problem $\text{Sat}(\mathcal{G}_F, \Phi)$.

\begin{itemize}
  \item Input: A marked graph $F$ and a graph property $\Phi \subseteq \mathbb{G}$.
  \item Question: “$\exists G \in \mathcal{G}_F : G \models \Phi$?”
\end{itemize}

Throughout this section, $F$ denotes a marked graph and $\mathcal{G}_F$ denotes the family of graphs defined in Definition 6. Sometimes, it will be crucial in the following that $F$ or $\Phi$ are fixed. We will clarify this by writing $\text{Sat}_F(\cdot)$ or $\text{Sat}_\Phi(\cdot)$, respectively.

For various well-studied graph properties, the satisfiability problem is always decidable. This includes problems where $\Phi$ states that a graph is planar (resp. is closed under graph-minors, resp. perfect, $k$-colorable, etc.). This is a direct consequence of the following fact.

**Theorem 4** Let either $\mathcal{G}_F$ be finite (i.e., a singleton) or $\Phi$ be any graph property which is closed under taking induced subgraphs (or both). Then, $\text{Sat}(\mathcal{G}_F, \Phi)$ is decidable.

**Proof** Since $F$ is an induced subgraph for every $G \in \mathcal{G}_F$, it is enough to check whether $F \models \Phi$. This is possible, because $\Phi$ is decidable by definition. □

In many cases, graph properties are expressible either in monadic second-order logic (MSO for short) or in first-order logic (FO for short). MSO is a rich and versatile class to define graph properties. Moreover, we allow quantification over both, sets of vertices and sets of

\begin{footnotesize}
12 The construction of the marked graphs changed perhaps the threshold, but not the period which is therefore still 1. For simplicity, the new threshold is still called $t$.

13 Recall that $\rho(w)$ is realized as a graph with a natural linear order on the vertices: we have $ab^c a \leq ab^d a \iff c \leq d$.
\end{footnotesize}
edges. Since $w \in G$ defines graphs with a linear order, we can express in MSO, for example, that the number of vertices is even. We use the following well-known results as a black box.

First, given an MSO sentence $\Phi$ and $k \in \mathbb{N}$, it is decidable whether there exists a graph of tree-width at most $k$ satisfying $\Phi$, see, e.g., [5, 6, 26]. As a second black box, we use Trakhtenbrot’s theorem [27]: on input of an FO sentence $\Phi$, it is undecidable whether there exists a finite graph satisfying $\Phi$.

**Remark 3** Trakhtenbrot’s theorem also holds in the following smaller family $B_t$ of finite bipartite graphs. More precisely, we mean the following. Let $t \in \mathbb{N}$ be any fixed constant. Then, $B_t$ denotes the family of connected finite bipartite graphs which have at least $t$ vertices. Then, on input of an FO sentence $\Phi$, it is undecidable whether there exists a graph in $B_t$ satisfying $\Phi$.

**Theorem 5** Let $\Phi$ be an MSO sentence. Then, $\text{Sat}(\cdot)$ is decidable for marked graphs $F = (V_F, E_F, \mu)$ as inputs where at most one endpoint of each edge is marked.

**Proof** The family $G_F$ yields a family of graphs of bounded tree-width. Indeed, the bag size is uniformly bounded by $|V_F|$. The result follows from the papers cited above. □

The next theorem shows in particular that the FO theory is undecidable if there is an edge where both endpoints are marked, using Trakhtenbrot’s theorem.

**Theorem 6** Let $F$ be a marked graph where both endpoints of some marked edge are marked. Then, $\text{Sat}(\cdot)$ is undecidable.

**Proof** Let $G$ be any finite connected bipartite graph with at least $n + 1$ vertices. According to Theorem 3, the family $G_F$ contains a graph $G'$ which is the disjoint union of $G$ and $F$. Let $\Psi$ be a first-order sentence which expresses that $F$ appears and that every vertex outside of $F$ is part of a connected component which has more vertices than $F$. As $F$ is fixed, $\Psi$ is also of constant size, with $O(n)$ many vertex variables. In particular, we can assume that $\Psi$ is of the form $\exists x_1, \ldots, x_n \Psi'$, this way fixing the vertices of $F$ in $G'$. Moreover, on input $\Phi \in \text{FO}$ we can construct another FO sentence $\Phi'$ such that $G \models \Phi$ if and only if there is a connected component $G'$ of $G$ such that $G' \models \Phi'$. Namely, $\Phi'$ speaks about the graph $G'$, disregarding the vertices of $F$. Let us denote by $\varphi_F(\Phi)$ the formula $\exists x_1, \ldots, x_n (\Psi' \land \Phi')$. Then, $\text{FO'} = \{ \varphi_F(\Phi) \mid \Phi \in \text{FO} \}$ is a subset of FO. This renders the satisfiability problem $\{ \varphi_F(\Phi) \models \text{FO'} \mid \exists G' \in G_F : G' \models \varphi_F(\Phi) \}$ to be undecidable by Trakhtenbrot’s theorem as stated in Remark 3. Indeed, if it was decidable, then we could decide whether $G \models \Phi$ for any $G \in B_{n+1}$. □

Some graph properties where the problem $\text{Sat}(G_F, \Phi)$ is trivially decidable are covered by the next theorem, including the problem whether $G_F$ contains a non-planar graph, and various parametrized problems like: “Is there some $(V_G, E_G) \in G_F$ with a clique bigger than $\sqrt{|V_G|}$?”.

**Theorem 7** Let $F$ be any marked graph and $\Phi$ be a non-trivial graph property such that $G \models \Phi$ if and only if there is a connected component $G'$ of $G$ such that $G' \models \Phi$. Then, the answer to the satisfiability problem $\text{Sat}(G_F, \Phi)$ is always “Yes” in the following two cases.

1. There is some marked self-loop.
2. The property $\Phi$ is true for some bipartite edge-graph and there is some marked edge where both endpoints are marked.

14 Questions like this one save us from discussing encodings of numbers as a second input parameter.
Properties of graphs specified by a regular language

Proof Since $\Phi$ is non-trivial, there is some finite graph $G$ modeling $\Phi$. If there is some marked self-loop, then there is some $K \in G_F$ such that $K$ is a disjoint union of $F$ and $G$ by Theorem 3. In the second case, we can choose $G$ to be a bipartite edge-graph. By assumption, $F$ contains a marked edge where both endpoints are marked. Again, there is some $K \in G_F$ such that $K$ is a disjoint union of $F$ and $G$, so that we can apply Theorem 3.

Example 5 lists a few graph properties which are not covered by Theorem 7, but nevertheless the satisfiability problem is decidable. Recall that graph $G = (V_G, E_G)$ possesses a Hamiltonian cycle if the cycle on $|V_G|$ vertices is a subgraph of $G$. A matching is a collection of edges of a graph such that no pair of these has any common vertices. A matching is perfect if it contains $|V_G|/2$ many edges. A set $T$ of vertices of a graph $G = (V_G, E_G)$ is called a dominating set \(^{15}\) if each vertex $u \in V_G$ has a vertex of $T$ in its closed neighborhood $N_G[u] = \{u\} \cup \{v \mid (u, v) \in E\}$. A set $D$ of vertices of a graph $G = (V_G, E_G)$ is called a defensive alliance in [16] if it is nonempty and each vertex $u \in D$ has at least half of its closed neighborhood within $D$.

Example 5 Let $F = (V_F, E_F, \mu)$ denote a marked graph as input. Then, the following problems are decidable.

1. Is there some $G \in G_F$ with a Hamiltonian cycle?
2. Is there some $G \in G_F$ with a perfect matching?
3. Is there some $G \in G_F$ with a dominating set of size at most $\log_2 |V_G|$?
4. Is there some $G \in G_F$ with a defensive alliance of size at most $\log_2 |V_G|$?

We explain these concrete examples one by one.

**Hamiltonian cycle.** In order to decide the existence of some $G \in G_F$ with a Hamiltonian cycle, we proceed as follows. Without restriction, we may assume that $F$ has no Hamiltonian cycle, because otherwise we are done. If there is any $G = (V_G, E_G) \in G_F$ with a Hamiltonian cycle $Z = (V_G, Z_G)$, then starting at any fixed vertex of $F$, the cycle yields a linear order on the vertices in $G$ and, by restriction, a linear order on $V_F$. Since $F$ is without any Hamiltonian cycle, the cycle leaves $F$ at some vertex $u_1 \in V_F$ and reenters $F$ at some vertex $v_1 \in V_F$. Continuing this way, we obtain a sequence of pairs $(u_1, v_1), \ldots, (u_k, v_k)$ with $1 \leq k \leq |V_F|$ before the cycle is closed. Let us look at the directed path $u_1 = w_0, w_1, \ldots, w_\ell = v_1$ on the cycle starting at some $u_i$ and ending in $v_i$ for some pair $(u_i, v_i)$ with $1 \leq i \leq k$. Suppose that $w_r = ab^i a$ and $w_s = ab^d a$ for $1 \leq r < s < \ell$ with $t \leq c < d$ and $[bc] = [bd]$. Then, we can modify the graph $G$ as follows: We remove all vertices $w_{r+1}, \ldots, w_s$ from $G$, and we introduce an edge $(w_r, w_{s+1})$. In this way, we obtain a smaller graph $G' \in G_F$ which still has a Hamiltonian cycle. Thus, if $G_F$ contains any graph with a Hamiltonian cycle, then $G_F$ contains such a graph with at most $p|V_F|$ vertices. Hence, it is enough to enumerate all graphs that have at most $p|V_F|$ vertices which have a Hamiltonian cycle and to check if any of them appears in $G_F$. Alternatively, one can enumerate all graphs in $G_F$ that have at most $p|V_F|$ vertices and check if they contain a Hamiltonian cycle.

**Perfect matching.** Let $V_F = \{x_1, \ldots, x_n\}$ and suppose that some $G = (V_G, E_G) \in G_F$ has a perfect matching. We have $V_F \subseteq V_G$. Hence, all $x_i \in V_F$ are matched by vertices $y_i \in V_G$. The induced subgraph $G[\{x_1, \ldots, x_n\}]$ has a perfect matching with at most $2|V_F|$ vertices. As in the precedent example, we enumerate and check all these graphs.

**Dominating set.** If there is no marked edge in $F = (V_F, E_F, \mu)$, then decide whether a dominating set with the desired property exists in $(V_F, E_F)$. In the second case, there is a

\(^{15}\) The notation $T$ refers to the German notion Träger.
marked vertex which is an endpoint of a marked edge. Then, $G_F$ contains a graph $G$ which is the (not disjoint) union of $F$ and an arbitrarily large star. The intersection of $F$ and the star is just one point. Thus, we find a graph $G$ in $G_F$ where $G$ has a dominating set of size $D(G)$ such that $D(G) \leq \log_2 |V_G|$. Actually, for every $\varepsilon > 0$, there is some $G \in G_F$ such that $D(G)/|V_G| < \varepsilon$. Thus, in the second case, we return “Yes.”

**Defensive alliance.** If $F = (V_F, E_F, \mu)$ contains no marked vertex at all, then $G_F = \{(V_F, E_F)\}$, so we have to check if $(V_F, E_F)$ contains a sufficiently small defensive alliance. Otherwise, we return “Yes.” Namely, in this case $G_F$ contains a graph $G$ that consists of $(V_F, E_F)$ plus $2|V_F|$ many isolated vertices. Now, one of these isolated vertices forms a sufficiently small defensive alliance by itself.

Frequently, we are not only interested in decision problems, but in computational problems. We illustrate this by computing the supremum (in $\mathbb{N}_\infty$) of the chromatic numbers over all the graphs in $G_F$. Recall that a graph $G = (V, E)$ is $k$-colorable if there is a function $c : V \rightarrow \{1, \ldots, k\}$ such that $(u, v) \in E \setminus \text{id}_V$ implies $c(u) \neq c(v)$. Indeed, self-loops should not have any influence on the chromatic number, because otherwise a graph with a self-loop could not be colored at all. For a finite graph, its chromatic number $\chi(G)$ is the minimal possible $k \in \mathbb{N}$ such that $G$ is $k$-colorable.

**Proposition 2** Let $F = (V_F, E_F, \mu)$ be a marked graph. Then, sup $\{\chi(G) | G \in G_F\} = \infty$ if and only if $F$ contains a marked self-loop. If $F$ is without any marked self-loop, then $\chi(V_F, E_F) = \max \{\chi(G) | G \in G_F\}$ is a natural number. Moreover, if $L \subseteq \mathbb{G}$ is regular, then sup $\{\chi(G) | G \in \rho(L)\}$ is effectively computable.

**Proof** Since $F \in G_F$, we have $\chi(V_F, E_F) \leq \sup \{\chi(G) | G \in G_F\}$. If $F = (V_F, E_F, \mu)$ has a marked self-loop, then $G_F$ contains for each $k \in \mathbb{N}$ a graph having a clique of size $k$ as a subgraph. Hence, sup $\{\chi(G) | G \in G_F\} = \infty$. Therefore, for the rest of the proof, we may assume that $F = (V_F, E_F, \mu)$ has no marked self-loops. For every $G \in G_F$, there is a graph morphism $\varphi : G \rightarrow (V_F, E_F)$. If $F = (V_F, E_F, \mu)$ has no self-loops at all, then we have $\chi(G) \leq \chi(F)$, because every fiber $\varphi^{-1}(v)$ is without any edge for $v \in V_F$. Thus, a $k$-coloring of $(V_F, E_F)$ induces a $k$-coloring of $G$. If there is a self-loop around a vertex $v$, then the loop is not marked by assumption. This loop is the only edge in $\varphi^{-1}(v)$, but, by definition, this has no influence on the chromatic number. □

Together with the results above, we have a meta-theorem for graph properties $\Phi$ with a decidable satisfiability problem, covering all cases where we have positive results.

**Theorem 8** Let $r : \mathbb{N} \rightarrow \mathbb{N}$ be a non-decreasing computable function and $\Phi$ be a graph property such that, for each marked graph $F$, the following property holds. If some graph $G = (V, E) \in G_F$ satisfies $\Phi$, then there is some graph $G' = (V', E') \in G_F$ such that $G'$ satisfies $\Phi$ and $G'$ has at most $r(|V_F|)$ vertices. Then, given as input a context-free grammar for a language $L \subseteq \mathbb{G}$ satisfying the $(b, t, p)$-torsion property, the following satisfiability problem

$$\text{Sat}(\rho(L), \Phi) = \exists G \in \rho(L) : G \models \Phi$$

is decidable.

**Proof** Since $L$ is context-free satisfying the $(b, t, p)$-torsion property, we find a regular language $R$ such that $\rho(R) = \rho(L)$. Splitting $R$ into finitely many cases, we are reduced to show the claim when the input is a single marked graph $F = (V_F, E_F, \mu)$. Taking $F$ as
input, we compute \( n = r(|V_F|) \) and we compute the list of all graphs with at most \( n \) vertices. Then, we check whether any graph in that list belongs to the family \( G_F \) and satisfies \( \Phi \), which is possible thanks to Corollary 1 (and as \( \Phi \) is decidable). \( \square \)

### 6 Conclusion and open problems

The starting point of our paper was the following idea: Decide a graph property \( \Phi \) not for a single instance as in traditional algorithmic graph theory, but generalize this question to a set of graphs specified by a regular language over a binary alphabet. “Let’s talk about a regular family of graphs, reader.” We chose a natural representation of graphs by words over a binary alphabet \( \Sigma \). Our results are rather robust, other “natural choices” work as well. Next, pick your favorite graph property \( \Phi \). For example, \( \Phi \) says that the number of vertices is a prime number. The property does not look very regular, there is no way to express the property, say, in MSO. Still, given a context-free language \( L \subseteq \Sigma^* \) which satisfies the \( b \)-torsion property and which encodes sets of graphs, we can answer the question if there exists a graph represented by \( L \) and which satisfies \( \Phi \). This is a consequence of Theorem 8 and Bertrand’s postulate that for all \( n \geq 1 \), there is a prime between \( n \) and \( 2n \).

Still, various problems are open. For example, is the satisfiability problem decidable for graph properties which are not covered by Theorem 8? This could mean that on input of a marked graph \((F, \mu)\), we can say “YES, there is such a graph in \( G_F \)” without producing a witness graph in \( G_F \) for this claim.

Another type of problems relates to model checking. Given a graph property \( \Phi \), we can define \( G(\Phi) = \{G \text{ is a finite graph} | G \models \Phi \} \). Suppose that \( \rho^{-1}(G(\Phi)) \) is regular. Given a regular language \( R \subseteq \Sigma^* \), can we decide whether \( G(\Phi) \subseteq \rho(R) \)? What about the equality \( G(\Phi) = \rho(R) \)? We can ask the same two questions if \( R \) is context-free.

Another area which we did not touch at all concerns complexity. We can state, however, an \( \text{NP} \) lower bound for \( \text{NP} \)-hard graph properties. Observe that our encoding of graphs by words is essentially optimal if we write exponents \( i \) which appear in factors \( ab^i a \) in binary. We let \( |ab^i a|_{\text{bin}} = 2 + \log_2(i) \), and this induces a binary length \( |w|_{\text{bin}} \) for \( w \in \mathbb{G} \) and also a natural binary length \( |F|_{\text{bin}} \) for marked graphs \( F = (V_F, E_F, \mu) \). If \( \Phi \) denotes an \( \text{NP} \)-hard graph property, then the problem \{\( F = (V_F, E_F, \mu) | \exists G \in G_F : G \models \Phi \)\} (with binary input size for \( F \)) is \( \text{NP} \)-hard. It is, however, not clear that it can be solved within \( \text{NP} \) assuming that property \( \Phi \) is in \( \text{NP} \).

**Acknowledgements** We thank Dietrich Kuske for pointing out that the formulation of [8, Thm. 3] is not correct as published in the proceedings. The correct statement is now Theorem 6. We also thank the anonymous referees of DLT’21 as well as of Acta Informatica for various suggestions to improve the presentation.

**Funding** Open Access funding enabled and organized by Projekt DEAL.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

---

16 A first song about a remotely similar theme was released in 1991 by Salt ‘n’ Pepa.
References

1. Anderson, T., Loftus, J., Rampersad, N., Santean, N., Shallit, J.: Detecting palindromes, patterns and borders in regular languages. Inf. Comput. 207, 1096–1118 (2009)

2. Anisimov, A.V.: Group languages. Kibernetika 4, 18–24 (1971). English translation in Cybernetics and Systems Analysis 4, 594–601 (1973)

3. Bera, S., Mahalingam, K.: Structural properties of word representable graphs. Math. Comput. Sci. 10, 209–222 (2016)

4. Bertelé, U., Brioschi, F.: Nonserial Dynamic Programming, vol. 91 of Mathematics in Science and Engineering. Academic Press, New York and London (1972)

5. Courcelle, B.: The expression of graph properties and graph transformations in Monadic Second-Order Logic. In: Rozenberg, G. (ed.) Handbook of Graph Grammars and Computing by Graph Transformations, vol. 1: Foundations, pp. 313–400. World Scientific (1997)

6. Courcelle, B., Engelfriet, J.: Graph Structure and Monadic Second-Order Logic: A Language-Theoretic Approach, vol. 138 of Encyclopedia of Mathematics and its Applications. Cambridge University Press (2012)

7. de Melo, A.A., de Oliveira Oliveira, M.: Second-order finite automata. In: Fernau, H. (ed.) Computer Science: Theory and Applications—15th International Computer Science Symposium in Russia, CSR 2020, Yekaterinburg, Russia, June 29–July 3, 2020, Proceedings, volume 12159 of Lecture Notes in Computer Science, pp. 46–63. Springer (2020)

8. Diekert, V., Fernau, H., Wolf, P.: Properties of graphs specified by a regular language. In: Moreira, N., Reis, R. (ed.) Developments in Language Theory—25th International Conference, DLT 2021, Porto, Portugal, August 16–20, 2021, Proceedings, vol. 12811 of Lecture Notes in Computer Science, pp. 117–129. Springer (2021)

9. Diestel, R.: Graph Theory, vol. 173 of Graduate Texts in Mathematics, 4th edn. Springer (2012)

10. Eilenberg, S.: Automata, Languages, and Machines, vol. A. Academic Press, New York and London (1974)

11. Ginsburg, S., Spanier, E.H.: Semigroups, Presburger formulas and languages. Pacific J. Math. 16, 285–296 (1966)

12. Güler, D., Krebs, A., Lange, K., Wolf, P.: Deciding regular intersection emptiness of complete problems for PSPACE and the polynomial hierarchy. In: Klein, S.T., Martin-Vide, C., Shapira, D. (eds.) Language and Automata Theory and Applications—12th International Conference, LATA 2018, Ramat Gan, Israel, April 9–11, 2018, Proceedings, vol. 10792 of Lecture Notes in Computer Science, pp. 156–168. Springer (2018)

13. Halin, R.: s-functions for graphs. J. Geom. 8, 171–186 (1976)

14. Kharlampovich, O.: The word problem for the Burnside varieties. J. Algebra 173, 613–621 (1995)

15. Kitaev, S., Seif, S.: Word problem of the Perkins semigroup via directed acyclic graphs. Order 25, 177–194 (2008)

16. Kristiansen, P., Hedetniemi, S.M., Hedetniemi, S.T.: Alliances in graphs. J. Comb. Math. Comb. Comput. 48, 157–177 (2004)

17. Kuske, D.: Second-order finite automata: expressive power and simple proofs using automatic structures. In: Moreira, N., Reis, R. (eds.) Developments in Language Theory—25th International Conference, DLT 2021, Porto, Portugal, August 16–20, 2021, Proceedings, vol. 12811 of Lecture Notes in Computer Science, pp. 242–254. Springer (2021)

18. Larsson, N.J., Moffat, A.: Off-line dictionary-based compression. Proc. IEEE 88, 1722–1732 (2000)

19. Lohrey, M., Maneth, S., Mennicke, R.: XML tree structure compression using RePair. Inf. Syst. 38, 1150–1167 (2013)

20. McNaughton, R.: Papert, Seymour: Counter-Free Automata. The MIT Press, Cambridge, Mass (1971)

21. Muller, D.E., Schupp, P.E.: Groups, the theory of ends, and context-free languages. J. Comput. Syst. Sci. 26, 295–310 (1983)

22. Parikh, R.J.: On context-free languages. J. ACM 13, 570–581 (1966)

23. Robertson, N., Seymour, P.: Graph minors. III. Planar tree-width. J. Comb. Theory 36, 49–64 (1984)

24. Rubtsov, A.A., Vyalyi, M.N.: Automata equipped with auxiliary data structures and regular realizability problems. In: Han, Y.-S., Ko, S-K. (eds.) Descriptional Complexity of Formal Systems—23rd IFIP WG 1.02, International Conference, DCFS 2021, Virtual Event, September 5, 2021, Proceedings, vol. 13037 of Lecture Notes in Computer Science, pp. 150–162. Springer (2021)

25. Schützenberger, M.P.: On finite monoids having only trivial subgroups. Inf. Control 8, 190–194 (1965)

26. Seese, D.: The structure of the models of decidable monadic theories of graphs. Ann. Pure Appl. Logic 53, 169–195 (1991)
Properties of graphs specified by a regular language

27. Trahtenbrot, B.A.: The impossibility of an algorithm for the decision problem for finite domains (in Russian). Doklady Akademii Nauk SSSR, New Series 70, 569–572 (1950). English Translation in American Mathematical Society, Translations 23, 1–5 (1963)

28. Vyalyi, M.N., Rubtsov, A.A.: On regular realizability problems for context-free languages. Probl. Inf. Transm. 51, 349–360 (2015)

29. Wolf, P.: Decidability of the Regular Intersection Emptiness Problem. Master’s thesis, Universität Tübingen, Germany (2018)

30. Wolf, P.: On the decidability of finding a positive ILP-instance in a regular set of ILP-instances. In: Hospodár, M., Jirásková, G., Konstantinidis, S. (eds.) Descriptional Complexity of Formal Systems—21st IFIP WG 1.02 International Conference, DCFS 2019, Košice, Slovakia, July 17–19, 2019, Proceedings, vol. 11612 of Lecture Notes in Computer Science, pp. 272–284. Springer (2019)

31. Wolf, P.: From decidability to undecidability by considering regular sets of instances. In: Cordasco, G., Gargano, L., Rescigno, A.A. (eds.) Proceedings of the 21st Italian Conference on Theoretical Computer Science, Ischia, Italy, September 14–16, 2020, vol. 2756 of CEUR Workshop Proceedings, pp. 33–46. CEUR-WS.org (2020)

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.