



Graph Automata and Graph Colorability

Antonios Kalampakas

College of Engineering and Technology, American University of the Middle East, Kuwait

Abstract. Automata recognizing graphs can be constructed by employing the algebraic structure of graphoids. For the construction of a graph automaton, the relations over the Kleene star of the state set must constitute a graphoid. Hence different kinds of graphoids produce graph automata with diverse operation and recognition capacity. In this paper we show that graph colorability is recognized by automata operating over the simplest possible abelian graphoid.

2020 Mathematics Subject Classifications: 68R10, 68Q45, 68Q70

Key Words and Phrases: Graph Theory, Graph Automata, Graph Colorability

1. Introduction

Automata on general (hyper)graphs were constructed for the first time in [5] by utilizing the algebraic properties of graphoids, i.e., magmoids satisfying the 15 equations of graphs which are specified in [2]. The notion of magmoids, introduced by Arnold and Dauchet in [1], is the algebraic structure employed to generate graphs from a finite set in a role similar to that of monoids for the generation of strings. A magmoid is a doubly ranked set equipped with two operations which are associative, unitary and mutually coherent in a canonical way (cf. [4] and [6]). In [7] Engelfriet and Vereijken proved that every graph with edges labeled over a finite doubly ranked set Σ can be built from a specific finite set of elementary graphs D , together with the elements of Σ , by using the operations of graph product and graph sum. From this result it is derived that graphs can be organized into a magmoid with operations product and sum. Although, as it was observed in [7], every hypergraph can be constructed in an infinite number of ways, this ambiguity was settled in [2] (see also [3]) by determining a *finite* set of equations \mathcal{E} with the property that two expressions represent the same hypergraph, if and only if, one can be transformed into the other through them.

Commencing from this result, a *graphoid* \mathbf{M} is defined as a magmoid with a designated set of elements that satisfy the equations \mathcal{E} . Hence $GR(\Sigma)$ can be structured into a graphoid by virtue of the set D of elementary graphs. The *relational magmoid* over a set Q is constructed by defining the operations of composition and sum on the set of relations

DOI: <https://doi.org/10.29020/nybg.ejpam.v16i1.4629>

Email addresses: antonios.kalampakas@aum.edu.kw (A. Kalampakas)

from Q^m to Q^n . This set is structured into a *relational graphoid* over Q , by specifying a set D of relations that satisfy the equations \mathcal{E} . A relational graphoid is called abelian when a particular relation of D consists of all the transpositions in Q . In [5] graph automata, with state set Q , were introduced by virtue of a specific abelian relational graphoid, denoted here by $\mathbf{TSRel}(\mathbf{Q})$, and by exploiting the fact that $GR(\Sigma)$ is the free graphoid generated by Σ . In the same paper it is postulated that different kinds of graphoids will produce graph automata with diverse operation and recognition capacity. In [8] it is shown that all abelian relational graphoids are characterized in the following way: a set Q generates an abelian relational graphoid if and only if Q is partitioned into disjoint abelian groups with operations univocally correlated with D . In other words it is proved that organizing $Rel(Q)$ in a relational graphoid is equivalent to partitioning the set Q and structuring every class in a group. The particular graphoid $\mathbf{TSRel}(\mathbf{Q})$ employed in [5] corresponds to the partitioning of Q into singleton sets each one being the trivial group. Hence, due to [8], it is the simplest possible abelian relational graphoid.

The smallest number of colors needed to color the vertices of the graph so that no two adjacent vertices share the same color is the chromatic number of the graph. A set of graphs is k -colorable if every graph of the set has chromatic number at most k .

In this paper we construct for every k , a graph automaton over the abelian graphoid $\mathbf{TSRel}(\mathbf{Q})$ which accepts all the k -colorable graphs. Hence it is manifested that graph automata are capable of recognizing important classes of graphs even when operating on the most trivial of the known graphoids. In the following section we recall basic definition for magmoids and hypergraphs. Graphoids are presented in Section 3 and the definition of a graph automaton is given. In Section 4 we construct the graph automaton that recognizes k -colorable graphs and illustrate its operation. Conclusions and future work are discussed in the last section.

2. Magmoids and Hypergraphs

A doubly ranked set $(A_{m,n})_{m,n \in \mathbb{N}}$ is a set A together with a function

$$rank : A \rightarrow \mathbb{N} \times \mathbb{N}$$

we set $A_{m,n} = \{a \in A \mid rank(a) = (m, n)\}$. In what follows we will drop the subscript and denote a doubly ranked set simply by $A = (A_{m,n})$. A *magmoid* is a doubly ranked set $M = (M_{m,n})$ equipped with two operations

$$\circ : M_{m,n} \times M_{n,k} \rightarrow M_{m,k}, \quad \square : M_{m,n} \times M_{m',n'} \rightarrow M_{m+m',n+n'},$$

which are associative in the obvious way, satisfy the distributivity law

$$(f \circ g) \square (f' \circ g') = (f \square f') \circ (g \square g')$$

whenever all the above operations are defined and are equipped with a sequence of constants $e_n \in M_{n,n}$, called units, such that

$$e_m \circ f = f = f \circ e_n, \quad e_0 \square f = f = f \square e_0, \quad e_m \square e_n = e_{m+n}$$

hold for all $f \in M_{m,n}$ and all $m, n \in \mathbb{N}$. Notice that, due to the last equation, the elements e_n are uniquely determined by e_1 . From now on e_1 will be simply denoted by e . The free magmoid $mag(\Sigma)$ generated by a doubly ranked set Σ is constructed in [2]. The sets $Rel_{m,n}(Q)$ of all relations from Q^m to Q^n

$$Rel_{m,n}(Q) = \{R \mid R \subseteq Q^m \times Q^n\}$$

can be structured into a magmoid with \circ being the usual relation composition and \square defined as follows: for $R \in Rel_{m,n}(Q)$ and $S \in Rel_{m',n'}(Q)$

$$R \square S = \{(u_1 u_2, v_1 v_2) \mid (u_1, v_1) \in R \text{ and } (u_2, v_2) \in S\},$$

where $u_1 \in Q^m, u_2 \in Q^{m'}, v_1 \in Q^n, v_2 \in Q^{n'}$. Notice that $Q^0 = \{\varepsilon\}$, where ε is the empty word of Q^* . The units are given by

$$e_0 = \{(\varepsilon, \varepsilon)\} \text{ and } e = \{(g, g) \mid g \in Q\}.$$

We denote by $Rel(Q) = (Rel_{m,n}(Q))$ the magmoid constructed in this way and call it the *relational magmoid of Q* .

An (m, n) -(hyper)graph $G = (V, E, s, t, l, begin, end)$ with edge labels from a doubly ranked set $\Sigma = (\Sigma_{m,n})$ is a tuple consisting of the set of nodes or vertices V , the set of edges E , the source and target functions $s : E \rightarrow V^+$ and $t : E \rightarrow V^+$ respectively, the labeling function $l : E \rightarrow \Sigma$ such that

$$rank(l(e)) = (|s(e)|, |t(e)|), \quad \text{for all } e \in E,$$

and the sequences of begin and end nodes $begin \in V^*$ and $end \in V^*$ with $|begin| = m$ and $|end| = n$. Notice that vertices can be duplicated in the begin and end sequences of the graph and also at the sources and targets of the edges. Isomorphism between two graphs is defined in the obvious way and we shall not distinguish between two isomorphic graphs. The set of all (m, n) -graphs over Σ is denoted by $GR_{m,n}(\Sigma)$ and we set

$$GR(\Sigma) = (GR_{m,n}(\Sigma))_{m,n \in \mathbb{N}}.$$

Ordinary unlabeled directed graphs are obtained as a special case of hypergraphs i.e., in the case that each hyperedge is binary (has one source and one target), every edge has the same label and the sequences $begin$ and end are the empty word.

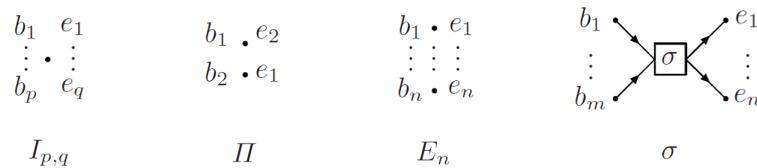
Given the (m, n) -graph $G = (V, E, s, t, l, begin, end)$ and the (n, k) -graph $H = (V', E', s', t', l', begin', end')$ then their *product* $G \circ H$ is the (m, k) -graph that is obtained by taking the disjoint union of G and H and then identifying the i^{th} end node of G with the i^{th} begin node of H , for every $i \in \{1, \dots, n\}$; also,

$$begin(G \circ H) = begin(G) \text{ and } end(G \circ H) = end(H).$$

The *sum* $G \square H$ of arbitrary graphs G and H is their disjoint union with their sequences of begin nodes concatenated and similarly for their end nodes (see [5, 8] for examples). For every $n \in \mathbb{N}$ we denote by E_n the discrete graph of rank (n, n) with nodes x_1, \dots, x_n and $begin = end = x_1 \cdots x_n$; we write E for E_1 . It is straightforward to verify that $GR(\Sigma) = (GR_{m,n}(\Sigma))$ with the operations defined above is a magmoid, whose units are the graphs E_n .

3. Graphoids and Graph Automata

In this section we present graph automata by employing the algebraic structure of graphoids as introduced in [5]. We denote by $I_{p,q}$ the discrete (p, q) -graph that has a single node x and whose begin and end sequences are $x \cdots x$ (p times) and $x \cdots x$ (q times) respectively, Π is the discrete $(2, 2)$ -graph that has two nodes x and y and whose begin and end sequences are xy and yx , respectively, also for every $\sigma \in \Sigma_{m,n}$, we denote again by σ the (m, n) -graph having only one edge and $m + n$ nodes $x_1, \dots, x_m, y_1, \dots, y_n$. The edge is labeled by σ , and the begin (resp. end sequence) of the graph is the sequence of sources (resp. targets) of the edge, viz. $x_1 \cdots x_m$ (resp. $y_1 \cdots y_n$).



Engelfriet and Vereijken, in [7], presented an algorithm that inductively constructs every graph $G \in GR(\Sigma)$ from the set $\Sigma \cup \{\Pi, I_{01}, I_{21}, I_{10}, I_{12}, \}$ by using graph product and graph sum. However, there are infinitely many ways to construct a given graph. This was overridden by identifying a finite set \mathcal{E} of equations with the property that two expressions represent the same graph if and only if one can be transformed into the other through these equations [2]. It is evident from this discussion that the equations \mathcal{E} are satisfied in $GR(\Sigma)$. Magmoids with such a property are called *graphoids*. Formally, a graphoid $\mathbf{M} = (M, D)$ consists of a magmoid M and a set $D = \{s, d_{01}, d_{21}, d_{10}, d_{12}\}$, with $s \in M_{2,2}$, $d_{\kappa\lambda} \in M_{\kappa,\lambda}$, such that the following equations hold:

$$s \circ s = e_2 \tag{1}$$

$$(s \square e) \circ (e \square s) \circ (s \square e) = (e \square s) \circ (s \square e) \circ (e \square s) \tag{2}$$

$$(e \square d_{21}) \circ d_{21} = (d_{21} \square e) \circ d_{21} \tag{3}$$

$$(e \square d_{01}) \circ d_{21} = e \tag{4}$$

$$s \circ d_{21} = d_{21} \tag{5}$$

$$(e \square d_{01}) \circ s = (d_{01} \square e) \tag{6}$$

$$(s \square e) \circ (e \square s) \circ (d_{21} \square e) = (e \square d_{21}) \circ s \tag{7}$$

$$d_{12} \circ (e \square d_{12}) = d_{12} \circ (d_{12} \square e) \quad (8)$$

$$d_{12} \circ (e \square d_{10}) = e \quad (9)$$

$$d_{12} \circ s = d_{12} \quad (10)$$

$$s \circ (e \square d_{10}) = (d_{10} \square e) \quad (11)$$

$$(d_{12} \square e) \circ (e \square s) \circ (s \square e) = s \circ (e \square d_{12}) \quad (12)$$

$$d_{12} \circ d_{21} = e \quad (13)$$

$$(d_{12} \square e) \circ (e \square d_{21}) = d_{21} \circ d_{12} \quad (14)$$

$$s_{m,1} \circ (p \square e) = (e \square p) \circ s_{n,1}, \quad \text{for all } p \in M_{m,n} \quad (15)$$

where $s_{m,1}$ is defined inductively by s and represents the graph associated with the permutation that interchanges the last n numbers with the first one [2]. We point out that the set of Equations (15) it only has to be valid for the elements of Σ in order to hold for every element of a magmoid generated by Σ [2]. Thus the pair $\mathbf{GR}(\Sigma) = (GR(\Sigma), D)$, with

$$D = \{\Pi, I_{01}, I_{21}, I_{10}, I_{12}\}$$

is a graphoid and in fact it is the free graphoid generated by Σ as it is illustrated in [5]. Given graphoids (M, D) and (M', D') , a magmoid morphism $H : M \rightarrow M'$ preserving D -sets, i.e., $H(s) = s'$ and $H(d_{\kappa\lambda}) = d'_{\kappa\lambda}$, is called a morphism of graphoids.

Graphoids constructed from the magmoid of relations $Rel(Q)$ over a given set Q are called *relational graphoids* and a relational graphoid is called abelian when

$$s = \{(g_1 g_2, g_2 g_1) \mid g_1, g_2 \in Q\}.$$

The abelian relational graphoid $\mathbf{TSRel}(Q) = (Rel(Q), D)$ that was used for the introduction of graph automata is constructed by setting s as above and

$$\begin{aligned} d_{01} &= \{(\varepsilon, g) \mid g \in Q\}, & d_{21} &= \{(gg, g) \mid g \in Q\}, \\ d_{10} &= \{(g, \varepsilon) \mid g \in Q\}, & d_{12} &= \{(g, gg) \mid g \in Q\}. \end{aligned}$$

A *nondeterministic relational graph automaton* is a structure

$$\mathcal{A} = (\Sigma, Q, \mathbf{Rel}(Q), \delta, I, T),$$

where Σ is the doubly ranked set of hyperedge labels, Q is the finite set of states, $\mathbf{Rel}(Q)$ is a relational graphoid over Q , $\delta : \Sigma \rightarrow \mathbf{Rel}(Q)$ is the doubly ranked transition function and I, T are initial and final rational subsets of Q^* . The function δ is uniquely extended into a morphism of graphoids

$$\bar{\delta} : GR(\Sigma) \rightarrow \mathbf{Rel}(Q),$$

where $\bar{\delta}(I_{\kappa\lambda}) = d_{\kappa\lambda}$ and $\bar{\delta}(\Pi) = s$, and the behavior of \mathcal{A} is given by

$$|\mathcal{A}| = \{F \mid F \in GR_{m,n}(\Sigma), \bar{\delta}_{\mathcal{A}}(F) \cap (I_{\mathcal{A}}^{(m)} \times T_{\mathcal{A}}^{(n)}) \neq \emptyset, m, n \in \mathbb{N}\}$$

where $I_{\mathcal{A}}^{(m)} = I_{\mathcal{A}} \cap Q^m$ and $T_{\mathcal{A}}^{(n)} = T_{\mathcal{A}} \cap Q^n$. From their construction, graph automata are finite machines due to the fact that the set of equations (1)-(15) is finite. A graph language is called recognizable whenever it is obtained as the behavior of a graph automaton. The class of all such languages over the doubly ranked set Σ is denoted by $Rec(\Sigma)$.

4. A Graph Automaton Recognizing k -colorable Graphs

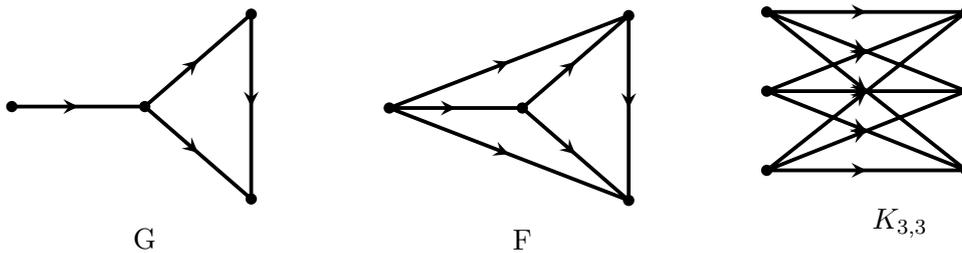
In this section we construct a relational graph automaton over the abelian graphoid $\mathbf{TSRel}(\mathbf{Q})$ recognizing k -colorable graphs.

For $k \in \mathbb{N}^*$, we set $\mathcal{A}_{clr}^k = (\Sigma, Q, \mathbf{TSRel}(\mathbf{Q}), \delta, I, T)$ with

- $\Sigma = \{a\}, rank(a) = (1, 1),$
- $Q = \{1, 2, \dots, k\},$
- $\delta(a) = \{(i, j) \mid i, j \in Q, i \neq j\},$
- $I = T = \{\varepsilon\}.$

It is clear that the automaton \mathcal{A}_{clr}^k reads unlabeled $(0, 0)$ -graphs with binary edges (one source and one target per edge), i.e., ordinary directed graphs.

As an example we shall illustrate the operations of $\mathcal{A}_{clr}^2, \mathcal{A}_{clr}^3$ and \mathcal{A}_{clr}^4 on the following graphs, where the label in every edge is a and thus omitted.



One of the representations of G is

$$G = I_{01} a I_{12} \begin{pmatrix} a \\ a \end{pmatrix} \begin{pmatrix} a \\ E \end{pmatrix} I_{21} I_{10}$$

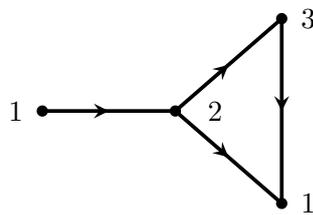
where graph product and graph sum are denoted, for simplicity, by horizontal and vertical concatenation. Note that since $GR(\Sigma)$ is the free graphoid, the operation of the automaton is the same for every representation of G . The consumption of G by \mathcal{A}_{clr}^3 gives

$$\bar{\delta}(G) = d_{01} \delta(a) d_{12} \begin{pmatrix} \delta(a) \\ \delta(a) \end{pmatrix} \begin{pmatrix} \delta(a) \\ e \end{pmatrix} d_{21} d_{10}$$

and an accepting sequence of transitions is

$$\{\varepsilon\} d_{01} \{1\} \delta(a) \{2\} d_{12} \begin{Bmatrix} 2 \\ 2 \end{Bmatrix} \begin{pmatrix} \delta(a) \\ \delta(a) \end{pmatrix} \begin{Bmatrix} 3 \\ 1 \end{Bmatrix} \begin{pmatrix} \delta(a) \\ e \end{pmatrix} \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} d_{21} \{1\} d_{10} \{\varepsilon\}$$

where the states are indicated in brackets. In graphical representation the states that the automaton reaches at each state are



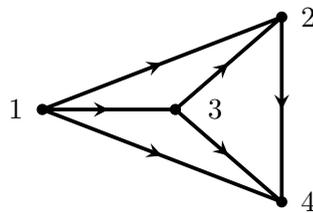
which is actually a proper 3-coloring of G . Similarly, a representation for F is

$$F = I_{01} I_{13} \begin{pmatrix} a \\ a \\ a \end{pmatrix} \begin{pmatrix} E \\ I_{12} \\ E \end{pmatrix} \begin{pmatrix} E \\ a \\ a \\ E \end{pmatrix} \begin{pmatrix} I_{21} \\ I_{21} \end{pmatrix} \begin{pmatrix} a \\ E \end{pmatrix} I_{21} I_{10}$$

and it is clear that there exists no successful transition of F by \mathcal{A}_{clr}^3 . A successful transition of F by \mathcal{A}_{clr}^4 is

$$\{\varepsilon\} d_{01} \{1\} d_{13} \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} \begin{pmatrix} \delta(a) \\ \delta(a) \\ \delta(a) \end{pmatrix} \begin{Bmatrix} 2 \\ 3 \\ 4 \end{Bmatrix} \begin{pmatrix} e \\ d_{12} \\ e \end{pmatrix} \begin{Bmatrix} 2 \\ 3 \\ 3 \\ 4 \end{Bmatrix} \begin{pmatrix} e \\ \delta(a) \\ \delta(a) \\ e \end{pmatrix} \begin{Bmatrix} 2 \\ 2 \\ 4 \\ 4 \end{Bmatrix} \\ \begin{pmatrix} d_{21} \\ d_{21} \end{pmatrix} \begin{Bmatrix} 2 \\ 4 \end{Bmatrix} \begin{pmatrix} \delta(a) \\ e \end{pmatrix} \begin{Bmatrix} 4 \\ 4 \end{Bmatrix} d_{21} \{4\} d_{10} \{\varepsilon\}$$

and the corresponding 4-coloring that is obtained by the state distribution of this transition is



Note that the elementary graph Π is not necessary for the representation of G and F . More generally, all planar graphs can be represented without employing Π . On the other hand, the non-planar graph $K_{3,3}$ is expressed as

$$K_{3,3} = \begin{pmatrix} I_{01} \\ I_{01} \\ I_{01} \end{pmatrix} \begin{pmatrix} I_{13} \\ I_{13} \\ I_{13} \end{pmatrix} \begin{pmatrix} a \\ \vdots \\ a \end{pmatrix} \Pi_s \begin{pmatrix} I_{31} \\ I_{31} \\ I_{31} \end{pmatrix} \begin{pmatrix} I_{10} \\ I_{10} \\ I_{10} \end{pmatrix}$$

where in the third parenthesis there are 9 a and Π_s stands for the graph that is associated with the permutation

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 1 & 4 & 7 & 2 & 5 & 8 & 3 & 6 & 9 \end{pmatrix}$$

Notice that as it is shown in [2] for every permutation we can construct, inductively by Π and E a graph that represents it. The graph $K_{3,3}$ is 2-colorable and an accepting transition of \mathcal{A}_{clr}^2 is

$$\{\varepsilon\} \begin{pmatrix} d_{01} \\ d_{01} \\ d_{01} \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \begin{pmatrix} d_{13} \\ d_{13} \\ d_{13} \end{pmatrix} \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \begin{pmatrix} \delta(a) \\ \vdots \\ \delta(a) \end{pmatrix} \begin{pmatrix} 2 \\ \vdots \\ 2 \end{pmatrix} d_s \begin{pmatrix} 2 \\ \vdots \\ 2 \end{pmatrix} \begin{pmatrix} d_{31} \\ d_{31} \\ d_{31} \end{pmatrix} \begin{pmatrix} 2 \\ 2 \\ 2 \end{pmatrix} \begin{pmatrix} d_{10} \\ d_{10} \\ d_{10} \end{pmatrix} \{\varepsilon\}$$

Hence, if we denote by $kCol$ the set of all k -colorable graphs, we obtain

Theorem 1. *For every $k \in \mathbb{N}$, the graph language $kCol$ is recognizable.*

5. Discussion

We showed that k -colorability is graph automaton recognizable even when the automaton operates over the most trivial abelian relational graphoid. This indicates that the graph automaton is a robust recognition mechanism and evokes numerous issues regarding the class of automaton recognizable graph languages. The introduced graph automata are nondeterministic in the sense that if $\sigma \in \Sigma_{m,n}$ then $\delta(\sigma)$ is a relation in $Q^m \times Q^n$. The deterministic version of this definition is obtained by requiring that $\delta(\sigma)$ is a function from Q^m to Q^n , i.e., $\delta(\sigma)$ is an element of $Funct(Q)$ which is a submagmoid of $Rel(Q)$ called the magmoid of functions [5]. It is interesting to compare the two classes and in particular to investigate the existence of a deterministic graph automaton recognizing k -colorable graphs. As it has been proved in [8] an infinite number of non-isomorphic abelian relational graphoids exists. Two questions that naturally arise concern the recognition capacity of the corresponding automata as well as the existence of non-abelian or non-relational graphoids.

Moreover, the time complexity for checking if a specific graph belongs to the behavior of a graph automaton grows polynomially as a function of the number of states $|Q|$ (see [5]). On the other hand, the complexity of the membership problem for a given graph automaton as a function of the size of the graph for both the deterministic and the nondeterministic

case has not been investigated. Such a result would classify automaton recognizable graph languages according to their computational complexity.

In a different approach, graph colorability can be investigated for directed fuzzy graphs, introduced in [9, 10] in relation with the existing concept of fuzzy graph coloring as in [11]. Additionally, graph recognizability via automata could also be applied in scheduling problems using vertex colorability as in [12].

References

- [1] A Arnold and M Dauchet. Théorie des magmoides. *RAIRO Theoret. Inform. Appl.*, 12:235–257, 1978.
- [2] S Bozapalidis and A Kalampakas. An axiomatization of graphs. *Acta Inform.*, 41:19–61, 2004.
- [3] S Bozapalidis and A Kalampakas. Automata on patterns and graphs. In *Proceedings of the 1st Conference on Algebraic Informatics*, pages 31–52, 2006.
- [4] S Bozapalidis and A Kalampakas. Recognizability of graph and pattern languages. *Acta Inform.*, 42:553–581, 2006.
- [5] S Bozapalidis and A Kalampakas. Graph automata. *Theoret. Comput. Sci.*, 393:147–165, 2008.
- [6] S Bozapalidis and A Kalampakas. Pattern language recognition and generation. *Pure Mathematics and Applications*, 22:1–38, 2011.
- [7] J Engelfriet and J Vereijken. Context-free graph grammars and concatenation of graphs. *Acta Informatica*, 34:773–803, 1997.
- [8] A Kalampakas. Graph automata: The algebraic properties of abelian relational graphoids. *Lecture Notes in Computer Science*, 7020:168–182, 2011.
- [9] A Kalampakas, S Spartalis, and L Iliadis. Syntactic recognizability of graphs with fuzzy attributes. *Fuzzy Sets and Systems*, 229:91–100, 2013. Theme: Computer Science.
- [10] A Kalampakas, S Spartalis, L Iliadis, and E Pimenidis. Fuzzy graphs: algebraic structure and syntactic recognition. *Artif. Intell. Rev.*, 42:479–490, 2014.
- [11] S. Munoz, M. T. Ortuno, J. Ramirez, and J. Yanez. Coloring fuzzy graphs. *Omega*, 33(3):211–221, 2005.
- [12] U Ufuktepe and G Bacak. Applying mathematica and webmathematica to graph coloring. *Future Generation Computer Systems*, 23(5):716–720, 2007.