Algebraic approach to fasciagraphs and rotagraphs *

Sandi Klavžar a,*,1, Janez Žerovnik b,2

a Department of Mathematics, PF, University of Maribor, Koroška cesta 160, 62000 Maribor, Slovenia
b Faculty of Technical Sciences, University of Maribor, Smetanova 17, 62000 Maribor, Slovenia

Received 18 February 1994; revised 18 January 1995

Abstract

An algebraic approach is proposed which can be used to solve different problems on fasciagraphs and rotagraphs. A particular instance of this method computes the domination number of fasciagraphs and rotagraphs in $O(\log n)$ time, where $n$ is the number of monographs of such a graph. Fasciagraphs and rotagraphs include complete grid graphs $P_k \square P_n$ and graphs $C_k \square C_n$. The best previously known algorithms for computing the domination number of $P_k \square P_n$ are of time complexity $O(n)$ (for a fixed $k$).

1. Introduction

The notion of a polygraph was introduced in chemical graph theory as a generalization of the chemical notion of polymers [3]. Polygraphs are of interest not only in chemistry, but grid graphs, for example, provide one of the most frequently used models of processor interconnections in multiprocessor VLSI systems [7]. An important class of polygraphs form fasciagraphs and rotagraphs. For example, complete grid graphs are fasciagraphs and Cartesian products of cycles are rotagraphs.

One of the main motivations for the present paper is a widely studied problem of determining the domination number of complete grid graphs and Cartesian products of cycles [5, 7, 9, 11–13, 18]. Despite considerable effort, only few formulas are known for the domination number of these graphs. Furthermore, proof techniques, at least for the time being, lead to rather lengthy proofs, cf. [5].

In general, problems related to domination in graphs are widely studied [10]. The problem of computing the domination number of grid graphs is NP-complete while the complexity is open for complete grid graphs, cf. [6, 10]. Hence it is worthwhile to look for algorithms that compute the domination numbers of these graphs.

* This work was supported in part by the Ministry of Science and Technology of Slovenia under the grants J1-7036/94 and J2-6192-0101-94.
* Corresponding author
1 Also at IMFM, Department of Mathematics, Ljubljana, Slovenia.
2 Also at IMFM, Department of Theoretical Computer Science, Ljubljana, Slovenia.

0166-218X/96/$15.00 © 1996 Elsevier Science B.V. All rights reserved
SSDI 0166-218X(95)00058-5
We consider finite undirected and directed graphs. A graph will always mean an undirected graph, a digraph will stand for a directed graph. \( P_n \) and \( C_n \) will denote the path on \( n \) vertices and the cycle on \( n \) vertices, respectively. An edge \( \{u, v\} \) of a graph will be denoted \( uv \) (hence \( uv \) and \( vu \) mean exactly the same edge). An arc from \( u \) to \( v \) in a digraph will be denoted \( (u, v) \).

Let \( G_1, G_2, \ldots, G_n \) denote the set of arbitrary, mutually disjoint graphs, and let \( X_1, X_2, \ldots, X_n \) be a sequence of sets of edges such that an edge of \( X_i \) joins a vertex of \( V(G_i) \) with a vertex of \( V(G_{i+1}) \). For convenience we also set \( G_0 = G_n, G_{n+1} = G_1 \) and \( X_0 = X_n \). This in particular means that edges in \( X_n \) join vertices of \( G_n \) with vertices of \( G_1 \). A polygraph

\[
\Omega_n = \Omega_n(G_1, G_2, \ldots, G_n; X_1, X_2, \ldots, X_n)
\]

over monographs \( G_1, G_2, \ldots, G_n \) is defined in the following way:

\[
V(\Omega_n) = V(G_1) \cup V(G_2) \cup \cdots \cup V(G_n),
\]

\[
E(\Omega_n) = E(G_1) \cup X_1 \cup E(G_2) \cup X_2 \cup \cdots \cup E(G_n) \cup X_n.
\]

For a polygraph \( \Omega_n \) and for \( i = 1, 2, \ldots, n \) we also define

\[
D_i = \{u \in V(G_i) \mid \exists v \in G_{i+1}: uv \in X_i\},
\]

\[
R_i = \{u \in V(G_{i+1}) \mid \exists v \in G_i: uv \in X_i\}.
\]

In general \( R_i \cap D_{i+1} \) need not be empty. A polygraph together with its sets \( D_i \) and \( R_i \) is schematically shown on Fig. 1.

Assume that for \( 1 \leq i \leq n \), \( G_i \) is isomorphic to a fixed graph \( G \) and that we have identified each \( G_i \) with \( G \). In addition, let the sets \( X_i, 1 \leq i \leq n \), be equal to a fixed edge set \( X \). Then we call the polygraph a rotagraph and denote it \( \omega_n(G; X) \). A fasciagraph \( \psi_n(G; X) \) is a rotagraph \( \omega_n(G; X) \) without edges between the first and the last copy of a monograph. Formally, in \( \psi_n(G; X) \) we have \( X_1 = X_2 = \cdots = X_{n-1} \) and \( X_n = \emptyset \).

Since in a rotagraph all the sets \( D_i \) and the sets \( R_i \) are equal, we will denote them by \( D \) and \( R \), respectively. The same notion will be used for fasciagraphs as well, keeping in mind that \( R_n \) and \( D_n \) are empty.

The Cartesian product \( G = H \square K \) of graphs \( H \) and \( K \) is the graph with vertex set \( V(G) = V(H) \times V(K) \). Vertices \((x_1, x_2)\) and \((y_1, y_2)\) are adjacent in \( H \square K \) if either \( x_1 y_1 \in E(H) \) and \( x_2 = y_2 \) or \( x_2 y_2 \in E(K) \) and \( x_1 = y_1 \). Note that \( P_k \square P_n = \psi_n(P_k; X), P_k \square C_n = \omega_n(P_k; X), C_k \square C_n = \omega_n(C_k; X) \) and \( C_k \square P_n = \psi_n(C_k; X) \), where \( X \) is
the matching defined by the identity isomorphism between two copies of $P_k$ and $C_k$, respectively.

The rest of the paper is organized as follows. In the next section a concept of a path algebra is introduced and an algorithm is proposed which can be used to solve different problems on fasciagraphs and rotagraphs in logarithmic time. In Section 3 we give an instance of the algorithm which computes the domination number of a fasciagraph and a rotagraph. This in particular implies that the domination number of a complete grid graph $P_k \square P_n$ can be obtained in $O(\log n)$ time for a fixed $k$. In the last section we briefly show how the same approach can be used to compute the independence number of fasciagraphs and rotagraphs and how to decide $k$-colorability of such graphs. We finally observe that the approach can also be extended to polygraphs but in this case the algorithms become linear. However, since polygraphs have bounded tree-width, linear algorithms on polygraphs are already known [1, 2].

2. Path algebras and the algorithm

In this section a general framework is proposed for solving different problems on the class of fasciagraphs and rotagraphs. The essence of the method is a computation of powers of matrices over certain semirings. We wish to remark that similar ideas are implicitly used in [8, 15]. Before giving the algorithm, a concept of path algebras is introduced. We follow the approach given in [4], see also [17, 19].

A semiring $\mathcal{P} = (P, +, \circ, 0, 1)$ is a set $P$ on which two binary operations, $+$ and $\circ$, are defined such that

(i) $(P, +)$ forms a commutative monoid with 0 as unit,
(ii) $(P, \circ)$ forms a monoid with 1 as unit,
(iii) operation $\circ$ is left- and right-distributive over operation $+$,
(iv) for all $x \in P$, $x \circ 0 = 0 = 0 \circ x$.

An idempotent semiring (for all $x \in P$, $x + x = x$) is called a path algebra. It is easy to see that a semiring is a path algebra if and only if $1 + 1 = 1$ holds. Examples of path algebras include (for more examples we refer to [4]):

$\mathcal{P}_1$: $(\mathbb{R} \cup \{\infty\}, \text{min}, +, \infty, 0)$,
$\mathcal{P}_2$: $(\mathbb{R} \cup \{-\infty\}, \text{max}, +, -\infty, 0)$,
$\mathcal{P}_3$: $(\{0, 1\}, \text{max}, \text{min}, 0, 1)$.

Let $\mathcal{P} = (P, +, \circ, 0, 1)$ be a path algebra and let $\mathcal{M}_n(\mathcal{P})$ be the set of all $n \times n$ matrices over $P$. Let $A, B \in \mathcal{M}_n(\mathcal{P})$ and define operations $A + B$ and $A \circ B$ in the usual way:

$$(A + B)_{ij} = A_{ij} + B_{ij},$$

$$(A \circ B)_{ij} = \sum_{k=1}^{n} A_{ik} \circ B_{kj}.$$
the unit matrix as units of the semiring.

Let \( P \) be a path algebra and let \( G \) be a labeled digraph, i.e., a digraph together with a labeling function \( \ell \) which assigns to every arc of \( G \) an element of \( P \). Let \( V(G) = \{v_1, v_2, \ldots, v_n\} \). The labeling \( \ell \) of \( G \) is extended to paths as follows. For a path \( Q = (x_{i_0}, x_{i_1}, x_{i_2}, \ldots, x_{i_{k-1}}, x_{i_k}) \) of \( G \) let

\[
\ell(Q) = \ell(x_{i_0}, x_{i_1}) \circ \ell(x_{i_1}, x_{i_2}) \circ \cdots \circ \ell(x_{i_{k-1}}, x_{i_k}).
\]

Let \( S_{ij}^k \) be the set of all paths of order \( k \) from \( x_i \) to \( x_j \) in \( G \) and let \( A(G) \) be the matrix defined by \( A(G)_{ij} = \ell(x_i, x_j) \) if \( (x_i, x_j) \) is an arc of \( G \) and \( A(G)_{ij} = 0 \) otherwise. Now we can state the following well-known result (see, e.g. [4, p. 99]):

**Theorem 2.1.** \( (A(G)^k)_{ij} = \sum_{Q \in S_{ij}^k} \ell(Q) \).

Let \( \psi_n(G;X) \) and \( \omega_n(G;X) \) be a fasciagraph and a rotagraph, respectively. Set \( W = D \cup R = D \cup R \) and let \( N = 2^{|W|} \). Define a labeled digraph \( F = F(G;X) \) as follows. The vertex set of \( F \) is formed by the subsets of \( W \), which will be denoted by \( C_i \); in particular we will use \( C_0 \) for the empty subset. An arc joins a subset \( C_i \) with a subset \( C_j \) if \( C_i \) is not in a "conflict" with \( C_j \). Here a "conflict" of \( C_i \) with \( C_j \) means that using \( C_i \) and \( C_j \) as a part of a solution in consecutive copies of \( G \) would violate a problem assumption. For instance, if we search for a largest independent set, such a conflict would be an edge between a vertex of \( C_i \) and a vertex of \( C_j \). Let finally \( \ell : E(F) \rightarrow P \) be a labeling of \( F \) where \( P \) is a path algebra on the set \( P \). The general scheme for our algorithm is the following:

**Algorithm 2.2.**

1. Select an appropriate path algebra \( P = (P, +, \circ, 0, 1) \).
2. Determine an appropriate labeling \( \ell \) of \( F(G;X) \).
3. In \( \mathcal{M}_n(P) \) calculate \( A(F)^n \).
4. Among admissible coefficients of \( A(F)^n \) select one which optimizes the corresponding goal function.

It is well known that Step 3 of the algorithm can be done in \( O(\log n) \) steps. Hence if we assume that the size of \( G \) is a given constant (and \( n \) is a variable), then the algorithm will run in \( O(\log n) \) time. However, the algorithm is useful for practical purposes only if the number of vertices of the monograph \( G \) is relatively small, since the time complexity is in general exponential in the number of vertices of the monograph \( G \).

3. Domination numbers of fasciagraphs and rotagraphs

A set \( S \) of vertices of a graph \( G \) is a dominating set if every vertex from \( V(G) \setminus S \) is adjacent to at least one vertex in \( S \). The domination number \( \gamma(G) \) is the smallest number of vertices in a dominating set of \( G \). For complete grid graphs, i.e., graphs
P_k \square P_n$, algorithms were given in [9] which for a fixed $k$ compute $\gamma(P_k \square P_n)$ in $O(n)$ time. We are going to present an algorithm that computes $\gamma$ of fasciagraphs and rotagraphs in $O(\log n)$ time.

Let $\psi_n(G;X)$ and $\omega_n(G;X)$ be a fasciagraph and a rotagraph, respectively. Let $C_i, C_j \in V(\mathcal{G}(G;X))$, i.e., $C_i, C_j \subseteq D \cup R$, and consider for a moment $\psi_3(G;X)$. Let $C_i \subseteq D_1 \cup R_1$ and $C_j \subseteq D_2 \cup R_2$, where $D_1 = D_2 = D$ and $R_1 = R_2 = R$ (cf. Fig. 2).

Let $\gamma_j(G;X)$ be the size of a smallest dominating set $S \subseteq G_2 \setminus((C_i \cap R_1) \cup (D_2 \cap C_j))$, such that $G_2$ is dominated by $C_i \cup S \cup C_j$. Then set

$$\ell(C_i, C_j) = |C_i \cap R| + \gamma_j(G;X) + |D \cap C_j| - |C_i \cap C_j|. \quad (1)$$

The labeling in particular implies that $(C_i, C_j)$ is an arc of $g(G;X)$ if $C_i \cap R \cap D \cap C_j = \emptyset$.

Recalling that $N = 2^{|P|}$ we now state:

Algorithm 3.1.

1. For a path algebra select $\mathcal{P}_1 = (\mathbb{R} \cup \{\infty\}, \min, +, \infty, 0)$.
2. Label $\mathcal{G}(G;X)$ as defined in (1).
3. In $\mathcal{H}_N(\mathcal{P}_1)$ calculate $A(\mathcal{G})^\eta$.
4. Let $\gamma(\psi_n(G;X)) = (A(\mathcal{G})^\eta)_{00}$ and $\gamma(\omega_n(G;X)) = \min_{i} (A(\mathcal{G})^\eta)_{ii}$.

Theorem 3.2. Algorithm 3.1 correctly computes $\gamma(\psi_n(G;X))$ and $\gamma(\omega_n(G;X))$ and can be implemented to run in $O(\log n)$ time.

Proof. The time complexity was already argued in the general case. We add here that for Step 2 of Algorithm 3.1 any procedure for computing the domination number can be used since the time complexity is clearly constant in $n$. With the same argument, Step 4 can be computed in constant time.

We next show the correctness of the algorithm. By Theorem 2.1,

$$(A(\mathcal{G})^\eta)_{00} = \min_{Q \in S_0^\infty} \ell(Q) = \min_{i_1, i_2, \ldots, i_{n-1}} (\ell(C_0, C_{i_1}) + \ell(C_{i_1}, C_{i_2}) + \cdots + \ell(C_{i_{n-1}}, C_0)).$$

Assume now that the minimum is attained on indices $i_1, i_2, \ldots, i_{n-1}$. Then

$$(A(\mathcal{G})^\eta)_{00} = (\gamma_{0,i_1} + |D_1 \cap C_{i_1}|) + (|C_{i_1} \cap R_1| + \gamma_{1,i_2} + |D_2 \cap C_{i_2}|) + \cdots + (|C_{i_{n-1}} \cap R_{n-1}| + \gamma_{n-1,0}).$$
By the definition of $\gamma_{ij}$, the above expression is the size of a dominating set of $\psi_n(G;X)$. On the other hand, a smallest dominating set of $\psi_n(G;X)$ gives rise to such an expression, thus $(A(\mathcal{D})^n)_{00}$ is the size of a smallest dominating set of $\psi_n(G;X)$.

The correctness argument for the domination number of $\omega_n(G;X)$ is analogous and we omit it. \(\square\)

**Corollary 3.3.** For a fixed $k$, one can obtain $\gamma(P_k \square P_n)$ and $\gamma(C_k \square C_n)$ in $O(\log n)$ time.

We conclude the section with a short overview of computational results. A straightforward implementation of the algorithm was tested on graphs $P_k \square P_n$, $P_k \square C_n$, $C_k \square C_n$ and $C_k \square P_n$ for $k = 2 - 5$ and for $n$ up to 1000. The known formulas for $P_k \square P_n$ and $C_k \square C_n$, \([5, 13]\), were used for checking the results. In one case the situation was opposite. Namely, it is proved in \([13]\) that for $n \geq 5$

$$\gamma(C_5 \square C_n) = \begin{cases} n & \text{if } n = 5k, \\ n + 1 & \text{if } n \in \{5k + 1, 5k + 2, 5k + 4\} \end{cases}$$

and that $\gamma(C_5 \square C_{5k+3}) \leq n + 2$. Our experiment showed that up to $n = 1000$ the upper bound $n + 2$ is the exact value of the domination number. It was then proved in \([18]\) that the formula is indeed $\gamma(C_5 \square C_{5k+3}) = n + 2$.

4. Some additional applications

The size of a largest independent set of vertices of a graph $G$ is called the independence number of $G$, $\alpha(G)$. Select $\mathcal{P}_2 = (\mathbb{R} \cup \{-\infty\}, \max, +, -\infty, 0)$ as a path algebra and define a labeling of $\mathcal{G}(G;X)$ similarly as in (1). The difference is that two vertices are in conflict (and hence the corresponding arc is labeled $-\infty$) if $(C_i \cap R) \cup (C_j \cap D)$ is not an independent set in $G$. Everything else is analogous as for the domination number, thus we have:

**Theorem 4.1.** One can compute $\alpha(\psi_n(G;X))$ and $\alpha(\omega_n(G;X))$ in $O(\log n)$ time.

As a second example, we consider the $k$-coloring problem. To solve it on fasciagraphs and rotagraphs, we first select $\mathcal{P}_3 = (\{0, 1\}, \max, \min, 0, 1)$ as a path algebra. We next define a labeled digraph $\mathcal{D}(G;X)$, slightly differently from how we did so far. The vertex set of $\mathcal{D}$ is formed by the $k$-colorings of $W = D \cup R$ or, equivalently, by the $k$-partitions of $W$ with parts being independent sets. An arc joins a $k$-coloring $C_i$ with a $k$-coloring $C_j$ if and only if the corresponding partitions coincide on their (possible) intersection in $G_2$ (cf. Fig. 2 again) and can be extended to a $k$-coloring of $G_2$. The labeling of $\mathcal{D}(G;X)$ is then defined just by the adjacency relation. Finally, in $\mathcal{H}_N(\mathcal{P}_3)$ we calculate $A(\mathcal{D})^n$ and conclude that $\psi_n(G;X)$ or $\omega_n(G;X)$ is $k$-colorable if and only if $(A(\mathcal{D})^n)_{00} = 1$ or $\max_i (A(\mathcal{D})^n)_{ii} = 1$, respectively. Thus we have:
Theorem 4.2. The k-coloring problem of the graphs \( \psi_n(G;X) \) and \( \omega_n(G;X) \) is solvable in \( O(\log n) \) time.

We finally add that the above approach for rotagraphs and fasciagraphs can be extended to polygraphs as well. Instead of computing a single graph \( \%_n(G;X) \) and calculating the \( n \)th power of \( A(\%) \), we must determine \( n \) graphs and calculate the matrix product of the corresponding matrices over an appropriate path algebra. This yields to \( O(n) \) algorithms for polygraphs. However, the tree-width of a polygraph can be bounded by a constant depending on the size of a monograph. (For definitions of a tree-width see, for example, \([16, 14]\), cf. also \([1]\).) Arnborg and Proskurowski \([2]\) (see also \([11]\)) obtained linear time algorithms for different problems of graphs with bounded tree-width, including dominating set, independent set and \( k \)-colorability problem. Their algorithms are linear in the size of the problem instance, but are exponential in the tree-width of the involved graphs – the case analogous to the present approach.

Concluding remarks

We learned from a referee that the domination number problem for \( k \times n \) grids, where \( k \) is fixed, has been claimed recently to have a constant time solution (Livingston and Stout, 25th International Conference on Combinatorics, Graph Theory and Computing, March 7–11, 1994, Florida Atlantic University). We would also like to add that recently several new formulas for the domination number of complete grid graphs have been established in \([20, 21]\). Finally, we wish to thank Martin Juvan for helpful remarks.

References