Discrete Applied Mathematics 208 (2016) 27–40

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

A constant time algorithm for some optimization problems in rotagraphs and fasciagraphs

M. Bouznif[a,b,1], J. Moncel[c,d], M. Preissmann[a,b,*]

[a] Univ. Grenoble Alpes, G-SCOP, F-38000 Grenoble, France
[b] CNRS, G-SCOP, F-38000 Grenoble, France
[c] CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France
[d] Univ. de Toulouse, Univ. Toulouse 1 Capitole, IUT Rodez, LAAS, F-31000 Toulouse, France

ARTICLE INFO

Article history:
Received 31 August 2015
Received in revised form 19 February 2016
Accepted 14 March 2016
Available online 23 April 2016

Keywords:
Combinatorial optimization problems
Constant-time algorithm
Grids
Fasciagraphs
Rotagraphs
Min-Plus algebra

ABSTRACT

This paper deals with optimization problems on rotagraphs and fasciagraphs. These graphs are repetitive structures that generalize grids and toroidal grids, respectively. We develop a theoretical framework and get linear-time and constant-time generic algorithms for wide classes of combinatorial problems which are defined in a purely combinatorial way. These classes of problems include in particular many classical optimization problems that are NP-complete in general. Our results unify in a single framework many results of the literature, in particular on the topic of domination – and its variants – in grids.

1. Introduction

Fasciagraphs and rotagraphs are graphs with a particular repetitive structure: They consist in a linear or circular sequence of copies of the same undirected graph, each copy being connected to the next one according to a fixed scheme.

An optimization problem in a graph consists often in the search of subsets of edges and/or vertices satisfying some property and optimizing a weight function. The case of the "minimum dominating set problem" will be used throughout this paper as an example. Our goal here is to determine when the repetitive structure of fasciagraphs and rotagraphs may be used to solve efficiently optimization problems on them. For this purpose we will need a characterization of "partial solutions", and a way to combine them in order to obtain a global solution.

In a previous paper [5], we addressed decision problems on fasciagraphs and rotagraphs. The present paper deals with general optimization problems, and therefore extends the theoretical framework developed in [5]. We show that, on fasciagraphs and rotagraphs, one can efficiently solve any combinatorial optimization problem that satisfies some well defined conditions. In this sense, our results generalize and improve those of Klavžar and Vesel [17], who developed a theoretical framework which is more restrictive than – and included in – ours. Taking advantage of the regular structure, we state conditions under which our algorithm computes in constant time a closed formula which allows to calculate the

* Corresponding author at: Univ. Grenoble Alpes, G-SCOP, F-38000 Grenoble, France.
E-mail addresses: marwane.bouznif@a-sis.com (M. Bouznif), julien.moncel@iut-rodez.fr (J. Moncel), myriam.preissmann@grenoble-inp.fr (M. Preissmann).
1 Present address: a-SIS, 8, rue de la Richelandière, F-42100 Saint Étienne, France.

http://dx.doi.org/10.1016/j.dam.2016.03.009

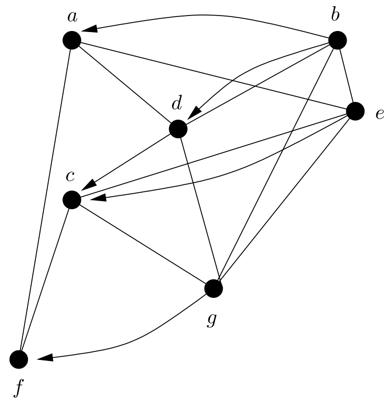

Fig. 1. Example of a mixed graph M.

optimal solution for all fasciagraphs and rotagraphs based on the same pattern. This can be seen as a generalization of the so-called "power method", described and developed in [24] to address domination problems, or of the distance-related invariants described in [15].

Many results of the literature dealing with dominating sets in grid-like structures rely on dynamic programming [1,11,18,20,24–26]. These results have been adapted for other combinatorial problems [7,9,13,14,16,17,23,27]. The framework developed in this paper provides a general argument proving that a large class of optimization problems can be efficiently solved by dynamic programming when restricted to fasciagraphs and rotagraphs. We moreover provide an explicit generic algorithm. Hence, our results unify all the above-mentioned papers, in the sense that the efficiency of dynamic programming for each of these problems is a simple consequence of our results.

In Section 2 we provide the definitions that we use in the sequel, in particular those of fasciagraphs and rotagraphs. Section 3 is dedicated to the case of rotagraphs, which is the easiest to describe thanks to the cyclicity of the structure (recall that rotagraphs generalize toroidal grids). This section contains the main ideas and results of the paper. The case of fasciagraphs is very similar to the one of rotagraphs, but requires some additional technical details. This case is indeed a little bit more complicated to describe, due to the "side effects" that we have to deal with. This case has been addressed separately in Section 4. In Sections 3.1 and 4.1 we recall the definitions of pseudo-local properties that were given in [5]. The notion of modular weight function is introduced in Sections 3.2 and 4.2. These definitions are important since the combinatorial problems that we are able to address are those corresponding to pseudo-local properties together with modular weight functions. We provide additional remarks and comments in Section 5, and the conclusion is then in Section 6.

2. Definitions and useful results

2.1. Graphs

A *mixed graph* M is a triple (V, E, A), where V is a set of elements called *vertices*, E is a set of 2-elements subsets of V called *edges* and A is a set of ordered pairs of elements of V called *arcs*. An arc (u, u) is called a loop. We denote an edge $\{u, v\}$ by uv or vu (see Fig. 1). Given a mixed graph M, the sets of vertices, edges and arcs of M will be denoted respectively by $V(M)$, $E(M)$ and $A(M)$.

A *directed graph* is a mixed graph $G = (V, E, A)$ where $E = \emptyset$; we denote it by $\vec{G} = (V, A)$. An *undirected graph* is a mixed graph $G = (V, E, A)$ where $A = \emptyset$; we denote it by $G = (V, E)$. In the following, the word "graph" will refer to an undirected graph, unless other specification.

Let k be a positive integer and $\vec{G} = (V, A)$ be a directed graph. A *path P of cardinality k* of $\vec{G}$, also called a *k-path*, is a sequence $v_1, \dots, v_{k+1}$ of (not necessarily distinct) vertices such that $(v_i, v_{i+1}) \in A$ for all $i \in \{1, \dots, k\}$. We then say that P is a path from v_1 to v_{k+1}.

A *circuit C of cardinality k*, also denoted *k-circuit*, of a directed graph $\vec{G} = (V, A)$ $(k \geq 1)$, is a path $v_1, \dots, v_{k+1}$ such that $v_1 = v_{k+1}$. The *cardinality of a path (or a circuit) Q* is denoted by $|Q|$.

Notice that what we call "path" (resp. "circuit") in this paper is often named "walk" (resp. "closed walk") in the literature.

A path or a circuit is said to be *elementary* if it does not contain more than once the same vertex (except for the first and last vertex in the case of circuits).

Given two paths $P = v_1, v_2, \dots, v_{k+1}$ and $Q = u_1, u_2, \dots, u_{l+1}$ such that $v_{k+1} = u_1$, the *concatenation* of P and Q is the $(k + l)$-path, denoted $P + Q$, which is equal to $v_1, v_2, \dots, v_{k+1} = u_1, u_2, \dots, u_{l+1}$.

A strongly connected component of a directed graph $\vec{G}$ is a maximal subgraph of $\vec{G}$ in which there exists a path from any vertex to any other vertex. A strongly connected component of a directed graph $\vec{G}$ is said to be *trivial* if it contains only one vertex and no arc.