Doctopic: Algorithms, automata, complexity and games ARTICLE IN PRESS TCS:11039

Theoretical Computer Science ••• (••••) •••–•••

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

On dynamic threshold graphs and related classes ☆

Tiziana Calamoneri *, Angelo Monti, Rossella Petreschi

Computer Science Department, "Sapienza" University of Rome, Italy

A R T I C L E   I N F O

Article history:
Received 12 May 2016
Received in revised form 30 November 2016
Accepted 4 January 2017
Available online xxxx

Keywords:
Fully dynamic graphs
Threshold graphs
Difference graphs
Chain graphs
Threshold signed graphs
Graph operations

A B S T R A C T

This paper deals with the well known classes of threshold and difference graphs, both characterized by *separators*, i.e. node weight functions and thresholds. We design an efficient algorithm to find the minimum separator, and we show how to maintain minimum its value when the input (threshold or difference) graph is fully dynamic, i.e. edges/nodes are inserted/removed. Moreover, exploiting the data structure used for maintaining the minimality of the separator, we study the disjoint union and the join of two threshold graphs, showing that the resulting graphs are threshold signed graphs, i.e. a superclass of both threshold and difference graphs. Finally, we consider the complement operation on all the three introduced classes of graphs.

All these operations produce in output the modified graph in terms of their separator and require time linear w.r.t. the number of different degrees. We observe that recomputing from scratch the separator would run either in linear (for threshold and difference graphs) or quadratic (for threshold signed graphs) time w.r.t. the number of nodes of the graph.

1. Introduction

In many applications of graph algorithms, *graphs* are *fully dynamic*, i.e. both edges and nodes may be inserted or eliminated.

Typically, one would like to answer to a precise query on the fully dynamic graph, so the goal is to update the data structure after dynamic changes, rather than to recompute it from scratch each time.

In this paper we deal with the maintenance of fully dynamic graphs when restricted to the classes of threshold and difference graphs.

Threshold graphs were introduced in 1977, independently as a model for three different problems: aggregation of inequalities in integer programming for set packing problems [7], node-labeling of the graph associated to a scheduling problem [11] and synchronization in parallel processing, lockout- and deadlock-free, solved with a semaphore-based approach [13]. After that, they have been defined many other times, as they are a natural model for a number of problems in many fields, e.g. resource allocation problems [20], scheduling [14], efficient parallel joins in relational databases [17], polyhedral combinatorics [5,4] and spectral graph theory [1].

The first definition of *difference graphs* – also known as *chain graphs* – goes back to 1972 [15] but they have been re-discovered in 1982 independently during an in depth analysis of threshold graphs [8] and to model a problem on partial

☆ Partially supported by the Italian Ministry of Education, University and Research, PRIN project "AMANDA: Algorithmics for MAssive and Networked DAta" and by Sapienza University of Rome.

* Corresponding author.
E-mail addresses: calamo@di.uniroma1.it (T. Calamoneri), monti@di.uniroma1.it (A. Monti), petreschi@di.uniroma1.it (R. Petreschi).

http://dx.doi.org/10.1016/j.tcs.2017.01.007
0304-3975/© 2017 Elsevier B.V. All rights reserved.

Please cite this article in press as: T. Calamoneri et al., On dynamic threshold graphs and related classes, Theoret. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.tcs.2017.01.007

orders [23]. Even these graphs have been then exploited in many fields such as recognizing poset dimension [23], studying the learning behavior of children [10], distributed memory multiprocessors [2], modeling software development process [9], dynamic networks [18] and cytoplasmic incompatibility in biology [19].

The pervasive application of threshold and difference graphs in so many fields makes natural to handle them in a fully dynamic way. To the best of our knowledge, few works deal with this topic. Namely, in [22] the problem of dynamically recognizing some classes of graphs (and among them threshold graphs) is handled. In [12] the authors consider the problem of adding/deleting edges with the aim of transforming a given graph into a threshold graph with the minimum number of changes. This paper is a contribution to the problem of the dynamic maintenance of threshold and difference graphs.

Among the numerous equivalent definitions of threshold and difference graphs, many exploit a node weight function and a threshold. This pair is called a *separator* and it is not unique. It is of interest to determine a *minimum separator*, i.e. a separator with minimum value of the threshold.

In this paper we present a new algorithm for finding a minimum separator. This algorithm, interesting by itself for its simplicity and linearity, is then considered as a pre-computation for maintaining the minimality after fully dynamically changing the input graph. To do this, we propose a simple data structure for maintaining the minimality of the separator, and handle some binary operations of two threshold graphs (disjoint union and join) whose result is in general not in the same graph class anymore, but in a superclass, called threshold signed graphs [3]; this superclass can be defined in terms of a node weight function and of two thresholds.

Finally, we tackle the problem of computing the weight function and the thresholds of the complement of one of these graphs without recomputing them from scratch, taking into account that the classes of threshold and threshold signed graphs are closed under complement while the complement of a difference graph is a threshold signed graph. To the best of our knowledge, it is the first time that the weight and thresholds of the complement of a given threshold signed graph is directly computed.

All the operations presented in this paper run in linear time w.r.t. the number of different degrees in the graph. This is particularly important, because recomputing from scratch the node weight function would require either linear (for threshold and difference graphs) or quadratic (for threshold signed graphs) time w.r.t. the number of nodes of the graph.

The paper is organized as follows: in Section 2 we recall some definitions. In Section 3, we describe the assignment algorithm determining a minimal integral separator for threshold and difference graphs. Moreover, we describe the data structures we use to store threshold and difference graphs; thanks to them, we are able to guarantee that all the operations handled in the next sections work in time that is linear w.r.t. the number of different degrees in the graph. Sections 4 and 5 describe how to add/delete an edge or a node, respectively, in either a threshold or a difference graph whenever it is possible to result in a graph of the same class (threshold or difference graphs). Section 6 describes a data structure feasible to store threshold signed graphs. In Section 7 we consider the operations of disjoint union and join of two threshold graphs and we show that the result is a threshold signed graph. In Section 8 we address the problem of adjusting the node weight function and the threshold(s) when the complement operation is applied to threshold, difference and threshold signed graphs, respectively. Finally, Section 9 concludes the paper.

2. Preliminaries

In this section, we list some definitions and properties useful for the rest of the paper. All the references of the results listed in this section can be found in the comprehensive survey book on threshold graphs, difference graphs and related topics by Mahadev and Peled [16].

Given a graph $G = (V, E)$, a subset of nodes $V' \subseteq V$ induces a *stable set* if $\forall u, v \in V'$ edge $(u, v) \notin E$; vice-versa, $V' \subseteq V$ induces a *clique* if $\forall u, v \in V'$ edge $(u, v) \in E$.

We denote by $\deg(v)$ the *degree* of node $v \in V$. A node v is *isolated* if $\deg(v) = 0$.

As we have already pointed out, there are many equivalent definitions of threshold graphs; in this paper we use the following:

Definition 1. A graph $G = (V, E)$ is a *threshold graph* if there is a mapping $a : V \to \mathbb{R}^+$ and a positive real number S such that

$$a(v) < S \text{ for all } v \in V \tag{1}$$

$$(v, w) \in E \text{ if and only if } a(v) + a(w) \geq S \tag{2}$$

The pair (a, S) will be called *separator* for graph G.

In Fig. 1.a a threshold graph with one of its separators is depicted.

Informally speaking, the nodes of a threshold graph can be partitioned into two sets, one inducing a clique and one a stable set; these sets are connected by a difference graph, that can be defined as follows:

Definition 2. A graph $G = (V, E)$ is a *difference graph* if there is a mapping $a : V \to \mathbb{R}$ and a positive real number T such that