Theoretical Computer Science 522 (2014) 1–12

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

(Nearly-)tight bounds on the contiguity and linearity of cographs

Christophe Crespelle [a,*], Philippe Gambette [b]

[a] Université Claude Bernard Lyon 1, DANTE/INRIA, LIP UMR CNRS 5668, ENS de Lyon, Université de Lyon, France
[b] Université Paris-Est, LIGM UMR CNRS 8049, Université Paris-Est Marne-la-Vallée, 5 boulevard Descartes, 77420 Champs-sur-Marne, France

ARTICLE INFO

Article history:
Received 17 August 2012
Received in revised form 7 July 2013
Accepted 26 November 2013
Communicated by D. Peleg

Keywords:
Contiguity
Linearity
Cographs
Graph encoding

ABSTRACT

In this paper we show that the contiguity and linearity of cographs on n vertices are both $O(\log n)$. Moreover, we show that this bound is tight for contiguity as there exists a family of cographs on n vertices whose contiguity is $\Omega(\log n)$. We also provide an $\Omega(\log n/\log\log n)$ lower bound on the maximum linearity of cographs on n vertices. As a by-product of our proofs, we obtain a min–max theorem, which is worth of interest in itself, stating equality between the rank of a tree and the minimum height of one of its path partitions.

1. Introduction

One of the most widely used operation in graph algorithms is the *neighborhood query*: given a vertex x of a graph G, one wants to obtain the list of neighbors of x in G. The classical data structure that allows to do so is the adjacency lists. It stores a graph G in $O(n+m)$ space, where n is the number of vertices of G and m its number of edges, and answers an adjacency query on any vertex x in $O(d)$ time, where d is the degree of vertex x.

This time complexity is optimal, as soon as one wants to produce the list of neighbors of x. On the other hand, in the last decades, huge amounts of data organized in the form of graphs or networks have appeared in many contexts such as genomics, biology, physics, linguistics, computer science, transportation and industry. In the same time, the need, for industrials and academics, to algorithmically treat this data in order to extract relevant information has grown in the same proportions. For these applications dealing with very large graphs, a space complexity of $O(n+m)$ is often very limiting. Therefore, as pointed out by [1], finding compact representations of a graph providing optimal time neighborhood queries is a crucial issue in practice. Such representations allow to store the graph entirely in memory while preserving the complexity of algorithms using neighborhood queries. The conjunction of these two advantages has great impact on the running time of algorithms managing large amount of data.

One possible way to store a graph G in a very compact way and preserve the complexity of neighborhood queries is to find an order σ on the vertices of G such that the neighborhood of each vertex x of G is an interval in σ. In this way, one can store the list of vertices of the graph in the order defined by σ and assign two pointers to each vertex: one toward its first neighbor in σ and one toward its last neighbor in σ. Therefore, one can answer adjacency queries on vertex x simply by listing the vertices appearing in σ between its first and last pointer. It must be clear that such an order on the vertices of G does not exist for all graphs G. Nevertheless, this idea turns out to be quite efficient in practice and some compression

* Corresponding author.
E-mail addresses: christophe.crespelle@inria.fr (C. Crespelle), philippe.gambette@univ-mlv.fr (P. Gambette).

0304-3975/$ – see front matter
http://dx.doi.org/10.1016/j.tcs.2013.11.036

techniques are precisely based on it [2,3]: they try to find orders of the vertices that group the neighborhoods together, as much as possible.

Then, a natural way to relax the constraints of the problem so that it admits a solution for a larger class of graphs is to allow the neighborhood of each vertex to be split in at most k intervals in order σ. The minimum value of k which makes possible to encode the graph in this way is a parameter called *contiguity* [4].

Another possible way of generalization is to use at most k orders $\sigma_1, \dots, \sigma_k$ on the vertices of G such that the neighborhood of each vertex is the union of exactly one interval taken in each of the k orders. This defines a parameter called the *linearity* of G [5] which is always less than or equal to the contiguity of G.

Only little is known about these two graph parameters. For example, the classes of graphs having contiguity (resp. linearity) at most k, where k is an integer greater than 1, have not been characterized, even for $k = 2$. Actually, the nature of these parameters seems to be quite different from those of the other classical graph parameters (e.g. based on decompositions), and it turns out that the classes of graphs known to have a nice behavior with regard to classical parameters, such as e.g. cographs (see Section 2), interval graphs and permutation graphs (see [6] for definitions of these classes), do not necessarily have a nice behavior with regard to contiguity and linearity. Thus, studying these two parameters is of key interest both for their practical implications and for their theoretical properties. In this paper, we aim at determining what is, in the worst case, the contiguity and linearity of cographs, that is, in other words, the maximum contiguity (resp. linearity) of cographs on n vertices.

Let us mention that in the following, all graphs are undirected, simple and loopless. For each of the two parameters we consider here, namely contiguity and linearity, there are actually two slightly different notions depending on whether one considers open neighborhoods (i.e. the set of neighbors of the vertex x, excluding the vertex x itself) or closed neighborhoods (i.e. the set of neighbors of the vertex x plus the vertex x). The corresponding notions are called *open contiguity* (resp. *open linearity*) and *closed contiguity* (resp. *closed linearity*). For contiguity, it does not make a big difference, as the open and closed parameters differ by at most one. For linearity, the situation is slightly different as it is not known whether the open linearity may exceed the closed linearity by more than one. But anyway, these two parameters are still equivalent in the sense that they differ at most by a multiplicative constant (at most two in this case). This is enough for us, as we consider the asymptotic behavior and we do not take into account multiplicative constants. Regarding the comparison between contiguity and linearity, it is straightforward to see that linearity is always less than contiguity (since one may duplicate the same order k times), but it is an open question to determine whether it can be significantly less for some graphs or not.

1.1. Related works

As we mentioned earlier, only little is known about contiguity and linearity of graphs. In the context of 0–1 matrices, [4,7] studied closed contiguity and showed that deciding whether an arbitrary graph has closed contiguity at most k is NP-complete for any fixed $k \geqslant 2$. For arbitrary graphs, [8] (Corollary 3.4) gave an upper bound on the value of closed contiguity which is $n/4 + O(\sqrt{n \log n})$.

Regarding graphs with bounded contiguity or linearity, only the class of graphs having closed contiguity 1 (or equivalently closed linearity 1) and the class of graphs having open contiguity 1 (or equivalently open linearity 1) have been characterized. The former is the class of proper (or unit) interval graphs [9], which is the subclass of interval graphs that admit a model whose intervals all have the same length. The latter class, i.e. graphs having *open contiguity* 1, is referred to as the class of biconvex graphs [6]. Biconvex graphs are a subclass of bipartite graphs properly containing bipartite permutation graphs, and which have, in terms of dimension theory for posets, dimension at most 3.

Finally, let us mention that [5] showed that both contiguity and linearity (closed and open) are unbounded for interval graphs as well as for permutation graphs, and that the four parameters can be up to $\Omega(\log n / \log\log n)$, where n is the number of vertices of the graph.

1.2. Our results

In this paper we show that, even for the class of cographs, the contiguity and the linearity are unbounded. Nevertheless, we show that they are both dominated by $O(\log n)$ for a cograph on n vertices. To this purpose, we show that the contiguity and linearity of a cograph G do not exceed the maximum height of a complete binary tree included (as a minor) in the cotree of G. As a by-product of our proof, we also establish a min–max theorem which is worth of interest in itself: the maximum height of a complete binary tree included (as a minor) in a tree T (known as the *rank* of tree T [10,11]) is equal to the minimum height of a *path partition* of T (see Definition 7).

Moreover, we exhibit a family of cographs $(G_n)_{n \in \mathbb{N}}$ on n vertices whose asymptotic contiguity is $\Omega(\log n)$, which implies that our $O(\log n)$ bound is tight. For the case of linearity, we exhibit a family of cographs whose asymptotic linearity is $\Omega(\log n / \log\log n)$. This leaves open the question of determining whether the linearity of cographs may be up to $\Omega(\log n)$ or not. In addition, it should be noted that we show that these lower bounds on the contiguity and linearity of cographs in the worst case are both reached on the families of cographs whose cotree is a complete binary tree. This emphasizes the question of knowing whether these two parameters are equivalent, for a cograph G, to some function of the maximum height of a complete binary tree included as a minor in the cotree of G.