Available online at www.sciencedirect.com
ScienceDirect
Discrete Applied Mathematics 156 (2008) 545–555
DISCRETE APPLIED MATHEMATICS
www.elsevier.com/locate/dam

Lower bounds on the pathwidth of some grid-like graphs☆

John Ellis[a], Robert Warren[b]

[a]Department of Computer Science, University of Victoria, PO Box 3055, Victoria, BC, Canada V8W 3P6
[b]School of Information Technology and Engineering, University of Ottawa, 800 King Edward Avenue, Ottawa, Ont., Canada K1N 6N5

Received 20 December 2005; received in revised form 27 December 2006; accepted 2 February 2007
Available online 3 March 2007

Abstract

We present proofs of lower bounds on the node search number of some grid-like graphs including two-dimensional grids, cylinders, tori and a variation we call “orb-webs”. Node search number is equivalent to pathwidth and vertex separation, which are all important graph parameters. Since matching upper bounds are not difficult to obtain, this implies that the pathwidth of these graphs is easily computed, because the bounds are simple functions of the graph dimensions. We also show matching upper and lower bounds on the node search number of equidimensional tori which are one less than the obvious upper bound.

Keywords: Pathwidth; Vertex separation; Node search number; Grids; Cylinders; Tori; Lower bounds

1. Introduction

Pathwidth is a graph parameter whose importance in determining the algorithmic complexity of width-bounded instances of intractable problems is well known. It is also well known that the related decision problem is NP-complete. In [8] it is shown that the problem remains NP-complete even for planar graphs of degree 3.

Here we examine some grid-like graph families, including two-dimensional grids, cylinders, tori and a variation thereof. We note that, although is it not hard to establish certain upper bounds on the pathwidth of these graphs, we do not know of proofs (except for grids, see below) that these bounds are also lower bounds. Once both lower and matching upper bounds are established and since the bounds are simple functions of the graph dimensions, it becomes easy to compute the pathwidth of these graph families.

The pathwidth of a graph is identical to its vertex separation [5] and the node search number is equal to vertex separation $+ 1$ [6]. Hence these concepts are equivalent. Here our proofs are worked exclusively in terms of node search number.

There is a substantial literature surrounding these concepts. Surveys can be found in [1,2,4]. The lower bound for grids is implied by the lower bound for the treewidth of grids shown in [3, Corollary 89]. Also, the concept of a “bramble/screen” developed in [9] could possibly be used to obtain the other lower bounds we prove here.

☆ This work was supported by the Natural Sciences and Engineering Research Council of Canada.
E-mail address: jellis@cs.uvic.ca (J. Ellis).

0166-218X/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.dam.2007.02.006

2. Definitions

Although the pathwidth of a graph, which we denote $pw(G)$, is perhaps the best known and widely applied of the three parameters listed in the Introduction, our proofs are worked entirely in terms of *node search number*, denoted $ns(G)$. We then use the equation $pw(G) = ns(G) - 1$ to obtain the theorems for the pathwidth. Informally, the search number of a graph is the minimum number of guards necessary to guarantee the capture of a fugitive who can move with arbitrary speed about the edges of the graph. Several versions of the game have been proposed. We are concerned here only with the so called *node search number*, which is defined as follows.

A *search step* is either the placing of a guard on a node, or the removal of a guard from a node. A *search sequence* is a sequence of search steps. Initially, all the edges of the graph are *contaminated*. A contaminated edge $\{u, v\}$ is cleaned whenever guards are simultaneously occupying both u and v. A clean edge will become recontaminated if the removal of a guard results in a path without guards from a contaminated edge to the clean edge. A *search strategy* S for a graph is a search sequence that results in all edges being simultaneously clean and no guards remaining on G. The cost of a strategy, $cost(S)$, is the maximum number of guards on G after any move in S. The node search number of a graph, denoted $ns(G)$, is $\min\{cost(S) \mid S \text{ is a strategy for } G\}$. In the rest of the paper we use "search number" to mean "node search number". When we say "at step t" we will mean "at the completion of the tth step".

A most useful result [6,7], assures us that we do not need to consider strategies that permit recontamination. No saving in guards can be gained in that way. We deduce that we need only consider strategies in which no guard on a node incident with a dirty edge is ever removed. This is because, first, if the guard is also incident with a clean edge, then it cannot be removed without causing recontamination and we need not consider recontaminating strategies. But if the guard is incident with nothing but dirty edges and is removed, the original placement was redundant. A strategy of no greater cost can be created by removing this pair of placement, removal moves.

We call guards on nodes incident with a dirty edge *critical* guards and the associated nodes *critical* nodes. It follows that if at some stage in a strategy all guards are critical, the next move must be a guard placement, and conversely, immediately prior to a guard placement, all nodes are critical. We will refer to this as the *criticality principle*.

The graph families we consider are variations on two-dimensional grids. A two-dimensional grid of height h and width w, which we call an (h, w)-grid, is the graph comprising the node set: $\{(x, y) \mid 0 \leqslant x < h, 0 \leqslant y < w\}$ and the edge set: $\{\{(u, v), (x, y)\} \mid |u - x| + |v - y| = 1\}$. Equivalently, the (h, w)-grid is the product of P_h and P_w where P_n is the simple path with n vertices. We assume in all cases that $w \geqslant 3$ and $h \geqslant 3$.

The (h, w)-cylinders are (h, w)-grids with wrap-around edges in the rows, i.e. the product of P_h and C_w where C_w is the simple cycle with w vertices. The (h, w)-tori are (h, w)-grids with wrap-around edges in both rows and columns, i.e. the product of C_h and C_w. We also consider what we call "(h, w)-orb-webs" which are a small variation on cylinders and are defined in their section.

These graphs and all the variations contain *rows*, all the nodes with the same first number in the vertex number pair, and *columns*, all the nodes with the same second number in the vertex number pair. We will say that a row or column is: *partly dirty* if some edges in that row or column are clean and some are dirty, *completely clean* if all edges are clean and *completely dirty* if all edges are dirty. A row or column will be called *almost clean* if the placement of one guard on some node in the row or column causes it to become completely clean.

3. Observations

We make a number of observations that we will use repeatedly in our proofs.

Observation 3.1. *A partly dirty column or row, with or without wrap-around edges, contains at least one critical guard, incident, respectively, with a dirty column or row edge, and it contains an unguarded node incident with a dirty edge.*

There must be a node incident with both a clean and a dirty edge. This node must be guarded. If all nodes incident with dirty edges are guarded then there are no dirty edges.

Observation 3.2. *A partly dirty column or row, with wrap-around edges contains at least two critical guards and two dirty edges, which share an unguarded node.*