Available online at www.sciencedirect.com
ScienceDirect
Discrete Applied Mathematics 156 (2008) 1822–1837
DISCRETE APPLIED MATHEMATICS
www.elsevier.com/locate/dam

Digraph searching, directed vertex separation and directed pathwidth

Boting Yang*,[1], Yi Cao

Department of Computer Science, University of Regina, Canada

Received 20 September 2006; received in revised form 7 May 2007; accepted 26 August 2007
Available online 24 October 2007

Abstract

The four digraph search models, directed search, undirected search, strong search, and weak search, are studied in this paper. There are three types of actions for searchers in these models: placing, removing, and sliding. The four models differ in the abilities of searchers and intruders depending on whether or not they must obey the edge directions when they move along the directed edges. In this paper, we investigate the relationships between these search models. We introduce the concept of directed vertex separation for digraphs. We also discuss the properties of directed vertex separation, and investigate the relations between directed vertex separation, directed pathwidth and search numbers in different search models.

Keywords: Digraph searching; Directed search; Undirected search; Strong search; Weak search; Directed vertex separation; Directed pathwidth

1. Introduction

Graph-searching problems serve as mathematical models for many real-world problems, such as capturing intruders in a building, clearing a complex system of interconnected pipes which is contaminated by some noxious gas, and killing a computer virus in a network system. In general, a *graph-* or *digraph-searching problem* is to find the minimum number of searchers to capture all the intruders hiding in a graph or digraph. Megiddo et al. [20] introduced the edge searching problem, in which there are three types of actions for searchers, i.e., placing, removing, and sliding, and an edge is cleared only by a sliding action in a proper way. Kirousis and Papadimitriou [18] introduced the node searching problem, in which there are two types of actions for searchers, i.e., placing and removing, and an edge is cleared if both end vertices are occupied by searchers. Bienstock and Seymour [7] introduced the mixed searching problem that combines the edge searching and node searching problems. LaPaugh [19] showed that recontamination of edges cannot reduce the number of searchers needed to clear a graph in the edge searching problem. There are several other graph-searching problems studied in [11–13,24,25]. A survey of graph-searching results can be found in [2,6,8,14].

Graph-searching problems were originally defined for undirected graphs. However, sometimes an undirected graph is not sufficient to represent all of the information of a real-world problem, for example, directed edges are required if

* Corresponding author.
E-mail addresses: boting@cs.uregina.ca (B. Yang), caoyi200@cs.uregina.ca (Y. Cao).
[1] Research was supported in part by NSERC and MITACS.

0166-218X/$ - see front matter
doi:10.1016/j.dam.2007.08.045

the graph models a road system with one-way streets. Johnson et al. [16] generalized the concepts of tree decomposition and treewidth to digraphs and introduced a digraph-searching problem accordingly. Recently, Adler [1] showed that this cops-and-robber game is not robber-monotone. Reed [22] defined another treewidth on digraphs. Safari [23] introduced d-width, which is related to the directed treewidth of a digraph. Evans and Safari [10] identified the class of digraphs whose directed treewidth and d-width are both equal to one. Berwanger et al. [5] and, independently, Obdržálek [21] introduced the DAG-width and the corresponding cops-and-robber game. Hunter and Kreutzer [15] introduced Kelly-width for digraphs. Alspach et al. [3] proposed four digraph search models in which searchers have only two types of actions, placing and sliding.

Barat [4] introduced a directed cops-and-robber game for digraphs. In this model, the robber hides only on vertices, and the cops stand also only on vertices. The robber can run from vertex u to vertex v along a directed path from u to v which does not contain any cop at a great speed at any time. The cops move by helicopters from vertex to vertex. The cops capture the robber if a cop lands on a vertex where the robber stands and all out-neighbors are occupied by cops. If the robber is invisible, the minimum number of cops needed to capture the robber hiding in a digraph D is the *cop number* of D, denoted by $\overline{\mathrm{cn}}(D)$.

Yang and Cao [26–28] introduced four digraph search models: directed search, undirected search, strong search and weak search, which are capable of being used as mathematical models for a wide range of real-world problems. These models differ in the abilities of the searchers and intruders depending on whether or not they must obey the edge directions as follows:

- In the *directed search* model, both searchers and intruders must follow the edge direction when they move along an edge.
- In the *undirected search* model, both searchers and intruders can move either from tail to head or from head to tail when they move alone an edge.
- In the *strong search* model, intruders must move in the edge directions but searchers can move either from tail to head or from head to tail when they move alone an edge.
- In the *weak search* model, searchers must move in the edge directions but intruders can move either from tail to head or from head to tail when they move alone an edge.

In [26–28], Yang and Cao proved that the directed, strong, and weak search models are monotonic, respectively.

Throughout this paper, we use D to denote a digraph that may have multiple edges, an ordered pair (u, v) to denote a directed edge with tail u and head v, and uv to denote an undirected edge with two end vertices u and v. Initially, all edges of digraph D are contaminated. In the directed or strong search model, each intruder can move from vertex u to vertex v along a directed path from u to v which does not contain any searcher at a great speed at any time; and in the undirected or weak search model, each intruder can move from vertex u to vertex v along an undirected path between u and v which does not contain any searcher at a great speed at any time. Each of the four search models has three types of actions for searchers: (1) placing a searcher on a vertex, (2) removing a searcher from a vertex, and (3) sliding a searcher along an edge. In particular, for the sliding action in the directed or weak search model, a searcher slides along an edge only from its tail to its head; and for the sliding action in the strong or undirected search model, a searcher slides along an edge from one end vertex to the other (ignoring the edge direction).

A *search strategy* is a sequence of actions such that the final action leaves all edges of D *uncontaminated* (or *cleared*). For the four search models, a contaminated edge can be cleared by a sliding action in a proper way.

- In the directed search model, a contaminated edge (u, v) can be cleared in one of the two ways by one sliding action: sliding a searcher from u to v along (u, v) while at least one searcher is located on u, or sliding a searcher from u to v along (u, v) while all edges with head u are already cleared.
- In the strong search model, a contaminated edge (u, v) can be cleared in one of the three ways by one sliding action: sliding a searcher from u to v along (u, v) while at least one searcher is located on u, sliding a searcher from u to v along (u, v) while all edges with head u are already cleared, or sliding a searcher from v to u along the edge (u, v).
- In the weak search model, a contaminated edge (u, v) can be cleared in one of the two ways by one sliding action: sliding a searcher from u to v along (u, v) while at least one searcher is located on u, or sliding a searcher from u to v along (u, v) while all edges incident with u except (u, v) are already cleared.