Graph Searching Tutorial

Fedor V. Fomin

A search game

is played on a graph between a group of searchers

and fugitive(s)

Exctenstions (digraphs, hypergraphs, polyhedra, sets...)

Many models of graph searching

- Searchers usually have no information on what fugitive plans to do
- The task is to identify the minimum number of searchers (or other resources) to guarantee successful search

Overview of the talk

- A bit of history
- Open problems:
  - Speed of players
  - Constrains on movements
  - How important is information

Disclaimer

- No way to mention everything
  - Distributed models
  - Self-stabilization
  - Cops & robbers, hunter and rabbits,
  - Directed graphs, submodular functions,
  - and much much more…

Prehistoric times

Breisch, R. (1967). An intuitive approach to speleotopology. Southwestern Cavers

An Intuitive Approach To SPELEOTOPOLOGY

— by Richard L. Breisch

INTRODUCTION

Speleotopology is the word I coined for the application of topology to the study of caves. To the best of my knowledge this is the first serious work on speleotopology.[1]

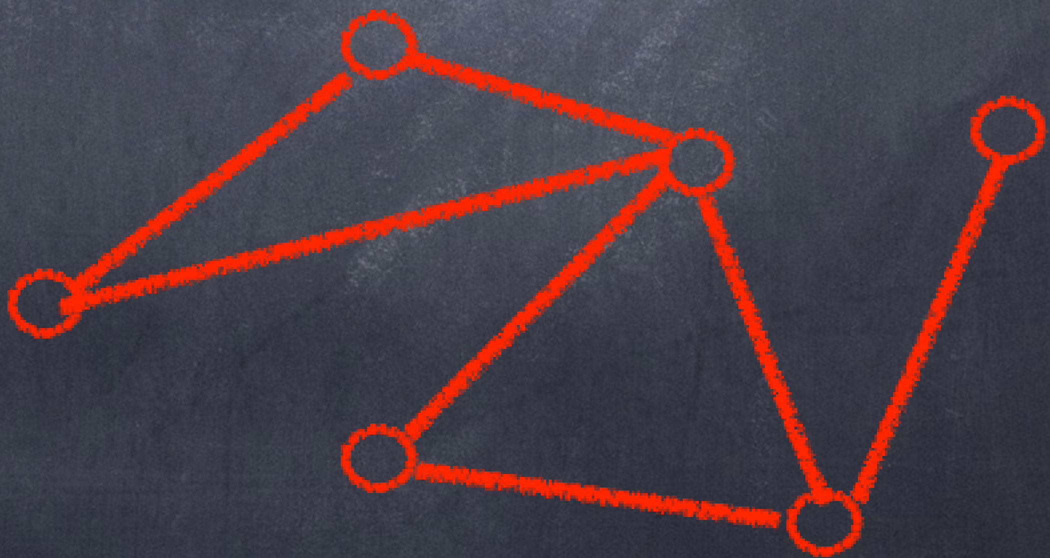
Topology is a branch of mathematics. Newman further defines it as "the geometry of distortion . It deals with fundamental geometric properties that are unaffected when we stretch, twist, or otherwise change an object's size and shape. It studies linear figures, surfaces, and solids....Its propositions hold as well for objects made of rubber as for the rigid figures encountered in (Euclidean) geometry."[2]

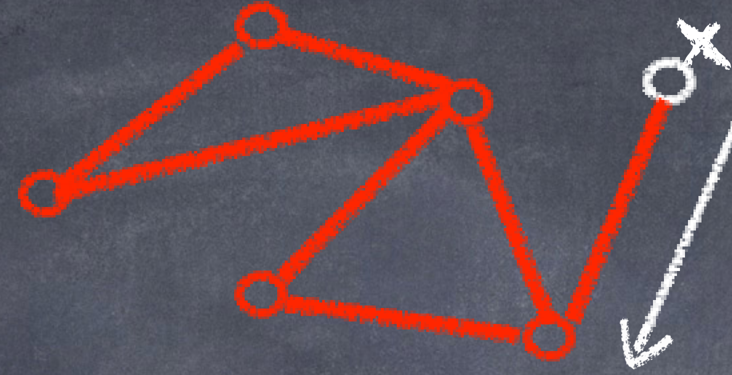
This paper presents two classical problems in topology plus an original problem which deals with the organization of a cave rescue party. I have chosen to present the concepts of speleotopology using an intuitive approach for two reasons: (1) Many of the spelunkers who read this will not have had sufficient background in mathematics to understand a more rigorous presentation and (2) I have never had a formal course in topology so therefore I

Cave Mythology: Rescue problem

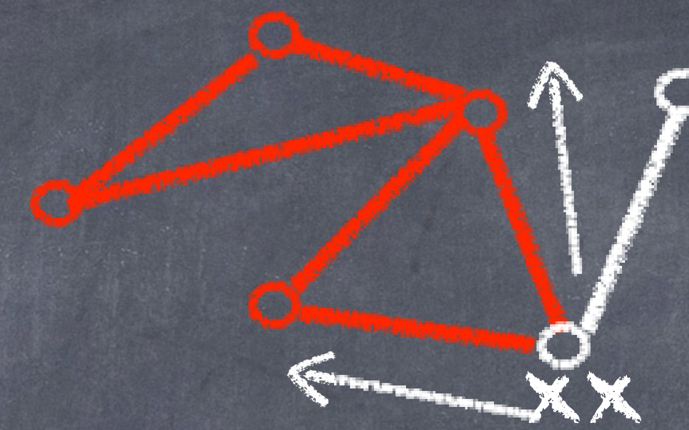
- A person is lost in a cave
- Find a minimum team of searchers sufficient for the rescue operation


I. CRETE
Chania
Akrotiri
Kissamos
Galatas
Georgioupoli
Rethimno
Ag. Pelagia
I. Dia
Iraklio
Perama
Samaria
Paleochora
Chersonissos
Mallia
Elounda
Ag. Nikolaos
Sitia
Kasteli
Kalo Chorio
Ag.Fotia
Plakias
Ag.Galini
Matala
Mirtos
Ierapetra

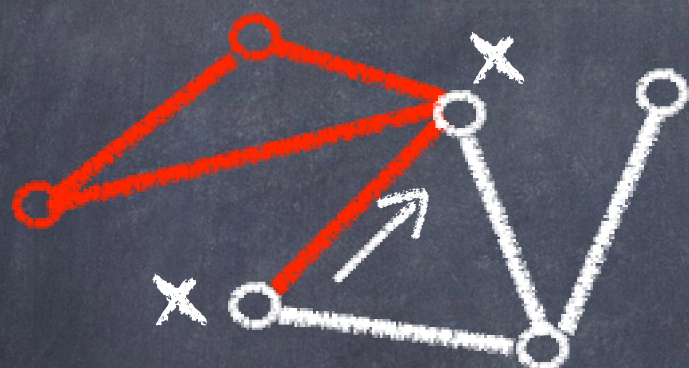




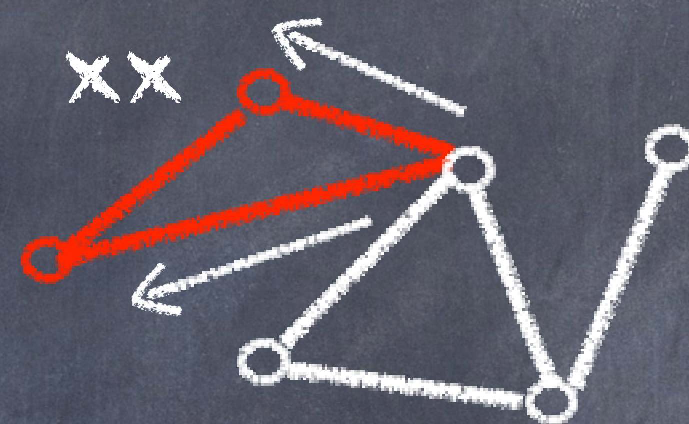
searchers: 1



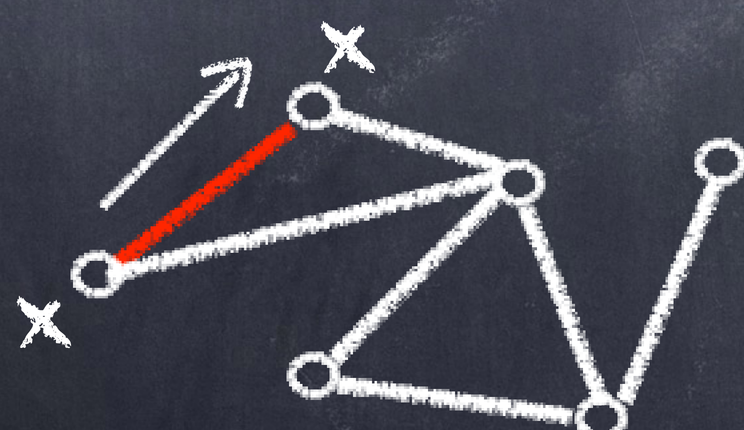
searchers: 2



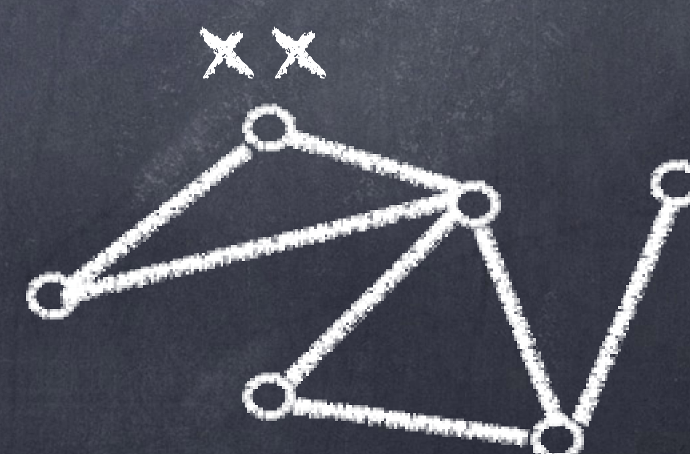
searchers: 2



searchers: 2



searchers: 2



searchers: 2

First mathematical papers on Graph Searching

Ancient times

Torrence Douglas Parsons (1976) Pursuit-evasion in a graph
Theory and Applications of Graph

PURSUIT-EVASION IN A GRAPH

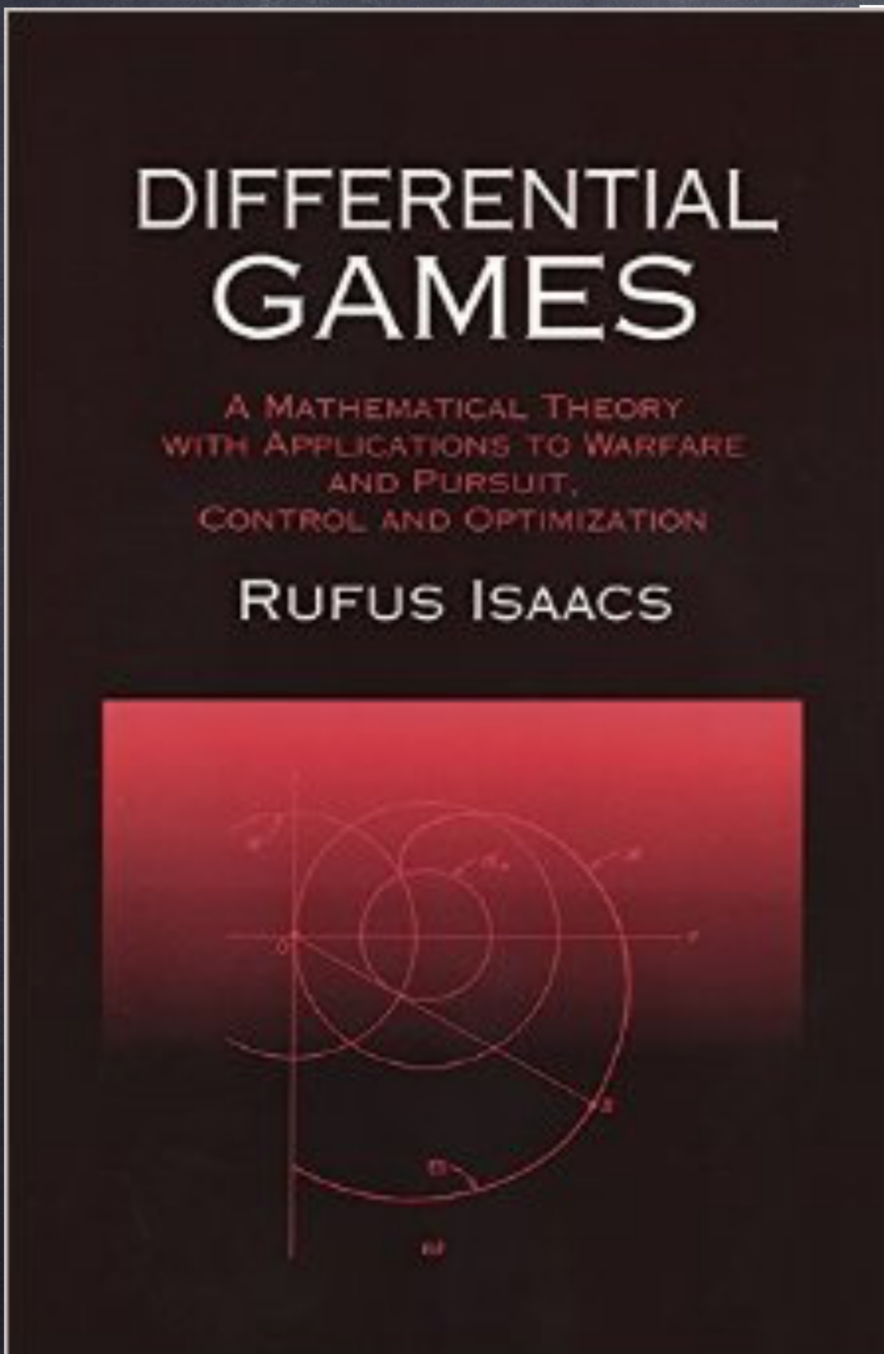
T.D. Parsons
The Pennsylvania State University

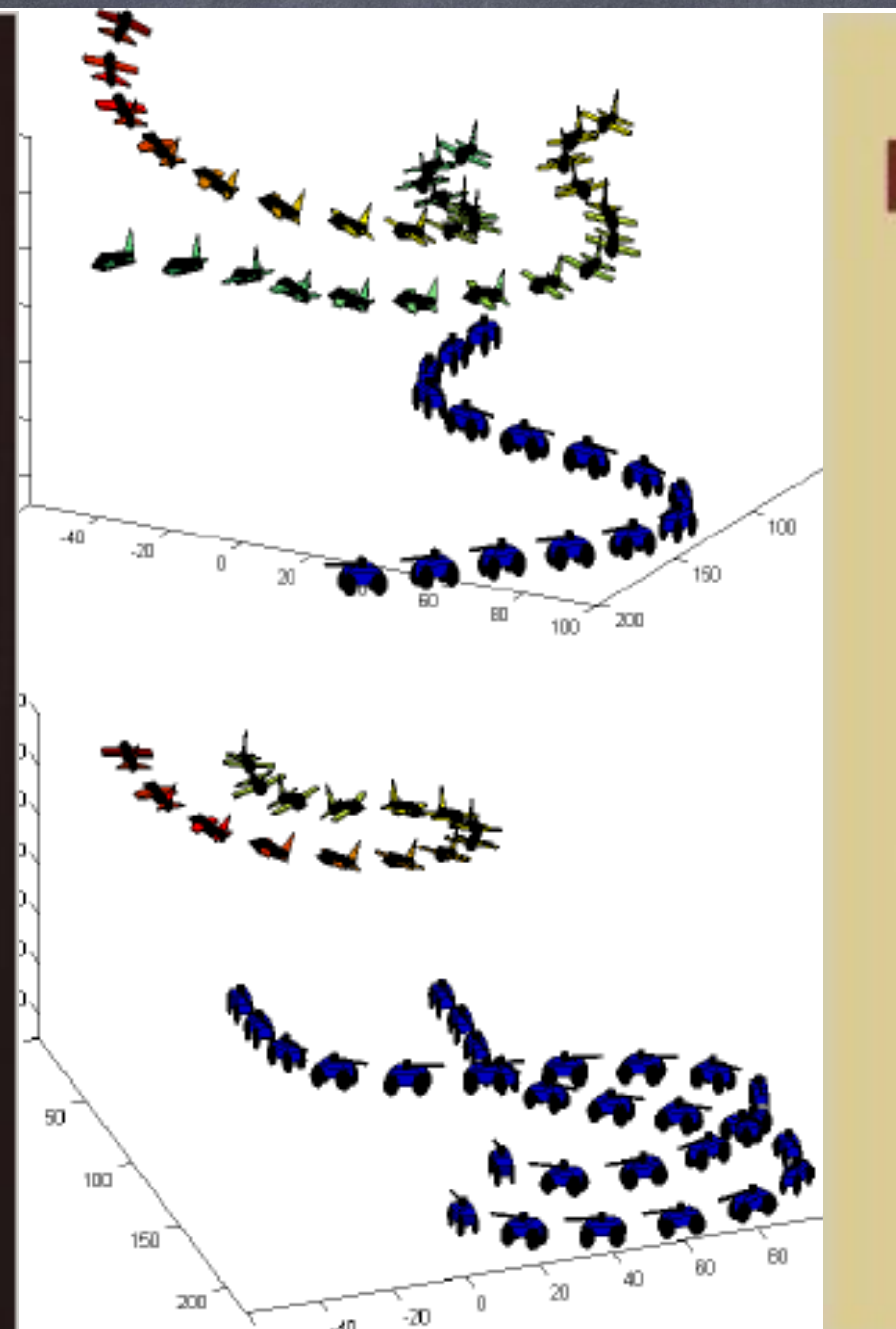
1. INTRODUCTION

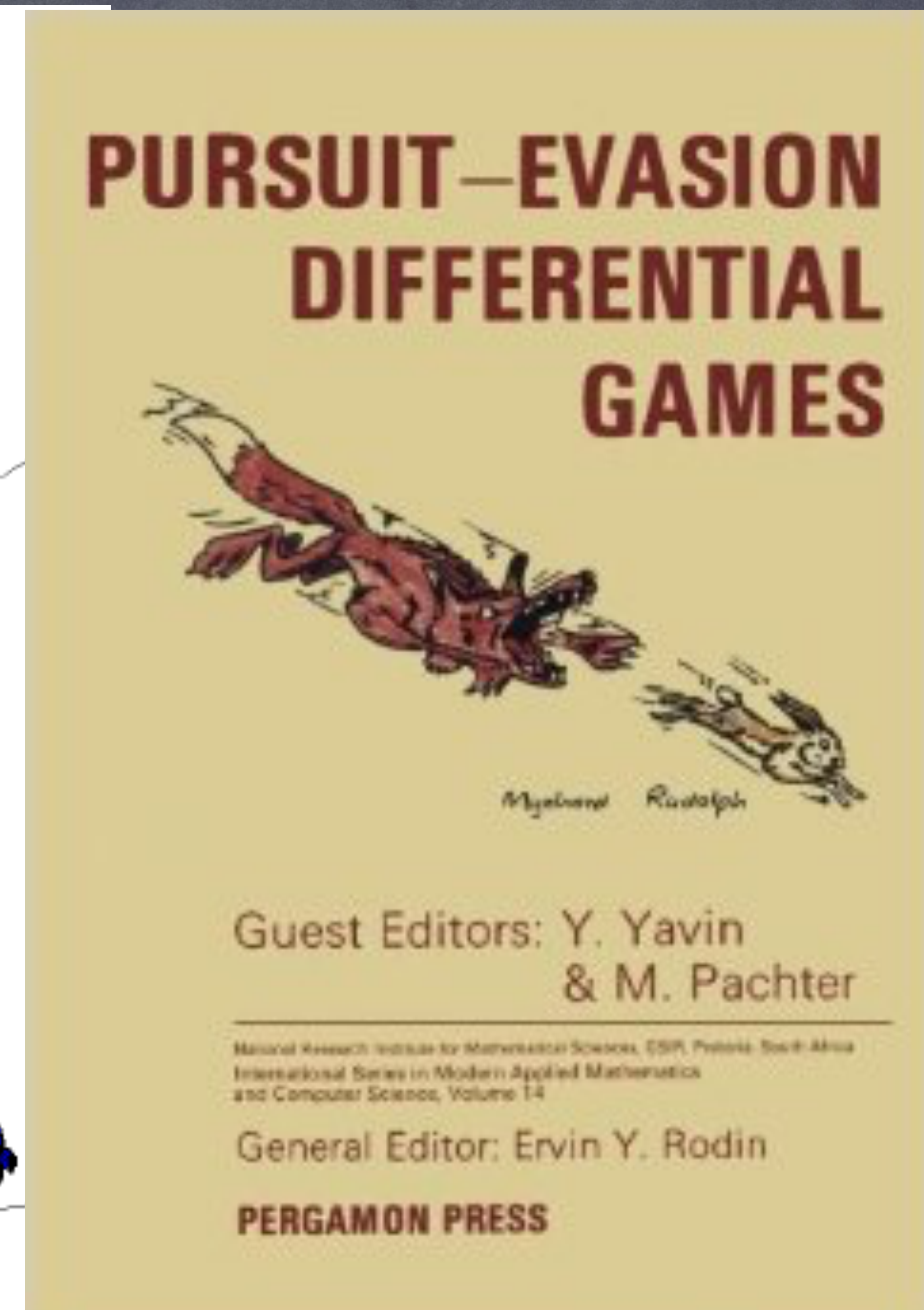
Suppose a man is lost and wandering unpredictably in a dark cave. A party of searchers who know the structure of the cave is to be sent to find him. What is the minimum number of searchers needed to find the lost man regardless of how he behaves?

This question was raised by my spelunker friend Richard Breisch, who developed informal arguments for many plausible conjectures about the problem. There are many inequivalent mathematical formulations of this problem, depending on the nature of the cave and the possible behavior allowed the searchers and the lost man. Breisch did not make precise which formulation he intended, although he gave numerous examples. One example was that of a circular cave, which requires two searchers;

Quite different motivation...







Petrov, Nikolaĭ Nikolaevich (1982)

AMERICAN MATHEMATICAL SOCIETY
MathSciNet
Mathematical Reviews

Previous | Up | Next

Citations
From References: 5
From Reviews: 3

MR661359 (84a:90099) 90D25 05C35 90B40
Petrov, N. N.
Some extremal search problems on graphs. (Russian)
Differentsial′nye Uravneniya **18** (1982), *no. 5,* 821–827, 916.

Author's summary: "We consider problems of pursuit on a graph in the absence of information about the evader and prove a number of theorems on the minimal number of pursuers necessary for successful completion of the pursuit."
{English translation: Differential Equations **18** (1982), no. 5, 591–595.}

Petrov, Nikolaĭ Nikolaevich (1982)

AMERICAN MATHEMATICAL SOCIETY
MathSciNet
Mathematical Reviews

Previous | Up | Next

Citations
From References: 8
From Reviews: 3

MR671157 (84b:90126) 90D26 05C35
Petrov, N. N.
A problem of pursuit in the absence of information on the pursued. (Russian)
Differentsial′nye Uravneniya **18** (1982), *no. 8,* 1345–1352, 1468.

The author considers a pursuit (search) problem with k pursuers and one mobile hider. The pursuers and the hider move in a finite connected graph Γ. It is assumed that the pursuers do not know the location of the hider. The main concern of the paper is to determine the least integer $\kappa(\Gamma)$ such that for $k \geq \kappa(\Gamma)$ the pursuit problem under consideration is solvable. The integer $\kappa(\Gamma)$ is a certain combinatorial-topological characteristic of the graph Γ. The author also introduces and investigates another combinatorial-topological characteristic of Γ which is called the degree of ramification. It turns out, and this is the main result of the paper, that the degree of ramification coincides with the integer $\kappa(\Gamma)$ provided that Γ is a tree. Related papers are by the author [same journal **18** (1982), no. 5, 821–827; MR0661359 (84a:90099)] and by T. D. Parsons [*Theory and applications of graphs* (Kalamazoo, Mich., 1976), 426–441, Lecture Notes in Math., 642, Springer, Berlin, 1978; MR0491364 (58 #10622)].

{English translation: Differential Equations **18** (1982), no. 8, 944–948 (1982).}

A. I. Subbotin

- Parsons and Petrov set a bit different definitions (continuous vs piece-wise linear vs discrete)
- Equivalence proved by Golovach in 1990
- Both started from trees and obtained combinatorial characterization of minimal search trees

Algorithmic questions

- Is the problem of deciding whether $s(G)<k$ in NP?
- Is the problem NP-hard?

J. ACM (1988)

NP-hard

The Complexity of Searching a Graph

N. MEGIDDO

Tel-Aviv University, Tel-Aviv, Israel

S. L. HAKIMI

Northwestern University, Evanston, Illinois

M. R. GAREY AND D. S. JOHNSON

AT&T Bell Laboratories, Murray Hill, New Jersey

AND

C. H. PAPADIMITRIOU

Massachusetts Institute of Technology, Cambridge, Massachusetts, and National Technical University of Athens, Athens, Greece

Abstract. T. Parsons originally proposed and studied the following pursuit-evasion problem on graphs: Members of a team of searchers traverse the edges of a graph G in pursuit of a fugitive, who moves along the edges of the graph with complete knowledge of the locations of the pursuers. What is the smallest number $s(G)$ of searchers that will suffice for guaranteeing capture of the fugitive? It is shown that determining whether $s(G) \leq K$, for a given integer K, is NP-complete for general graphs but can be solved in linear time for trees. We also provide a structural characterization of those graphs G with $s(G) \leq K$ for $K = 1, 2, 3$.

Categories and Subject Descriptors: F.2.2 **[Analysis of Algorithms and Problem Complexity]**: Nonnumerical Algorithms and Problems—*computations on discrete structures*; *sorting and searching*

General Terms: Algorithms, Theory, Verification

Additional Key Words and Phrases: NP-completeness, pursuit and evasion

Is the problem NP-complete?

- Yes, due to monotonicity: recontamination does not help to search a graph!

NP-complete

J. ACM (1993)

Recontamination Does Not Help to Search a Graph

ANDREA S. LAPAUGH

Princeton University, Princeton, New Jersey

Abstract. This paper is concerned with a game on graphs called *graph searching*. The object of this game is to clear all edges of a contaminated graph. Clearing is achieved by moving searchers, a kind of token, along the edges of the graph according to clearing rules Certain search strategies cause edges that have been cleared to become contaminated again. Megiddo et al. [9] conjectured that every graph can be searched using a minimum number of searchers without this recontamination occurring, that is, without clearing any edge twice. In this paper, this conjecture is proved. This places the graph-searching problem in NP, completing the proof by Megiddo et al. that the graph-searching problem is NP-complete. Furthermore, by eliminating the need to consider recontamination, this result simplifies the analysis of searcher requirements with respect to other properties of graphs.

Categories and Subject Descriptors: F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems—*computations on discrete structures*; G.2.2 [**Discrete Mathematics**]: Graph Theory—*graph algorithms*

General Terms: Algorithms, Theory

Additional Key Words and Phrases: Graph searching, NP-completeness, pursuit and evasion

simpler proof

JOURNAL OF ALGORITHMS **12**, 239–245 (1991)

Monotonicity in Graph Searching

D. BIENSTOCK* AND PAUL SEYMOUR

Bellcore, 445 South Street, Morristown, New Jersey 07960

Received May 1988; accepted November 1990

We give a new proof of the result, due to A. LaPaugh, that a graph may be optimally "searched" without clearing any edge twice.

1. INTRODUCTION

Let us regard a graph as a system of tunnels containing a (lucky, invisible, fast) fugitive. We desire to capture this fugitive by "searching" all edges of the graph, in a sequence of discrete steps, while using the fewest

Relation to width parameters

- Node search number (G) = pathwidth(G)+1
- Pathwidth(G)= min max clique in interval completion -1

Discrete Mathematics 55 (1985) 181–184 181
North-Holland

INTERVAL GRAPHS AND SEARCHING

Lefteris M. KIROUSIS
Department of Mathematics, University of Patras, Patras, Greece, and Department of Computer Science, National Technical University, Athens, Greece

Christos H. PAPADIMITRIOU
Department of Mathematics, University of Patras, Patras, Greece, and Department of Computer Science, National Technical University, Athens, Greece

Received 3 November 1983
Revised 5 September 1984

The interval thickness of a graph G is the minimum clique number over any interval supergraph of G. The node-search number is the least number of searchers required to clear the 'contaminated' edges of a graph. The clearing is accomplished by concurrently having searchers on both of its endpoints.
We prove that for any graph, these two parameters coincide.

Information Processing Letters 42 (1992) 345–350 24 July 1992
North-Holland

The vertex separation number of a graph equals its path-width

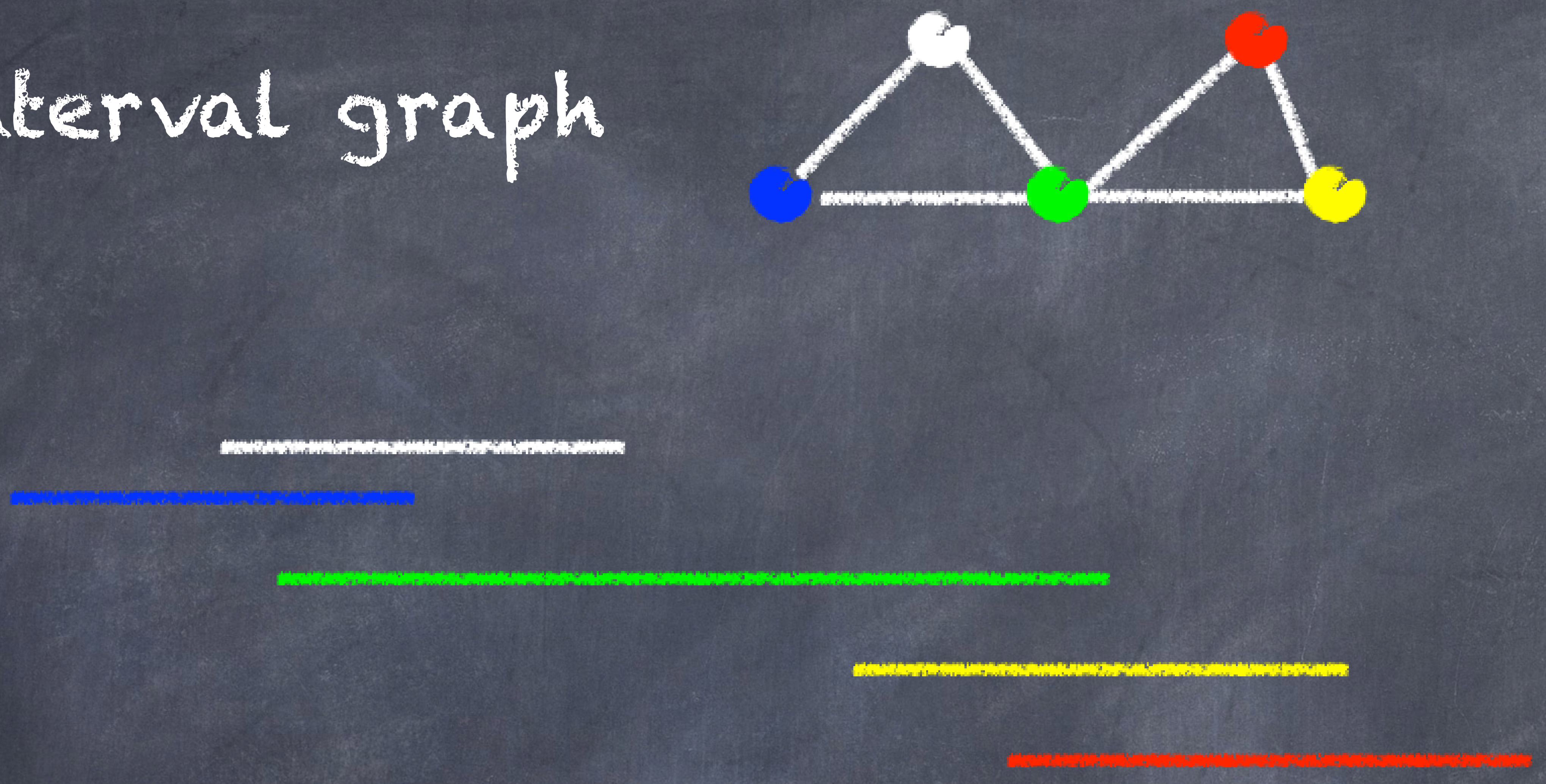
Nancy G. Kinnersley *
Department of Computer Science, University of Kansas, Lawrence, KS 66045, USA

Communicated by D. Dolev
Received 18 July 1990
Revised 2 January 1992

Abstract

Kinnersley, N.G., The vertex separation number of a graph equals its path-width, Information Processing Letters 42 (1992) 345–350.

Interval graph



How to node-search interval graph?

Relation to width parameters

- Helicopter search number (G) = treewidth(G)+1
- Treewidth(G)= min max clique in chordal completion -1

JOURNAL OF COMBINATORIAL THEORY, Series B **58**, 22–33 (1993)

Graph Searching and a Min–Max Theorem for Tree-Width

P. D. SEYMOUR

Bellcore, 445 South Street, Morristown, New Jersey 07962

AND

ROBIN THOMAS*

School of Mathematics, Georgia Institute of Technology, Atlanta, Georgia 30332

Received April 19, 1989

The tree-width of a graph G is the minimum k such that G may be decomposed into a "tree-structure" of pieces each with at most $k+1$ vertices. We prove that this equals the maximum k such that there is a collection of connected subgraphs, pairwise intersecting or adjacent, such that no set of $\leqslant k$ vertices meets all of them. A corollary is an analogue of LaPaugh's "monotone search" theorem for cops trapping a robber they can see (LaPaugh's robber was invisible).

Graph Theory of graph searching

- Strong relation to structural Graph Theory
- Treewidth, pathwidth, directed treewidth...

Algorithmic

- While Graph Searching is NP-complete
- It is in P on trees
- Deciding if k searchers suffice for fixed k is in P, moreover, it is FPT, i.e. f(k)poly(n) algorithms are known

But what about graph searching?

- constrains on dynamics of players
- properties of cleaned area
- available information
- bounds
- algorithms
- monotonicity/time issues

So what if fugitive is not that fast?

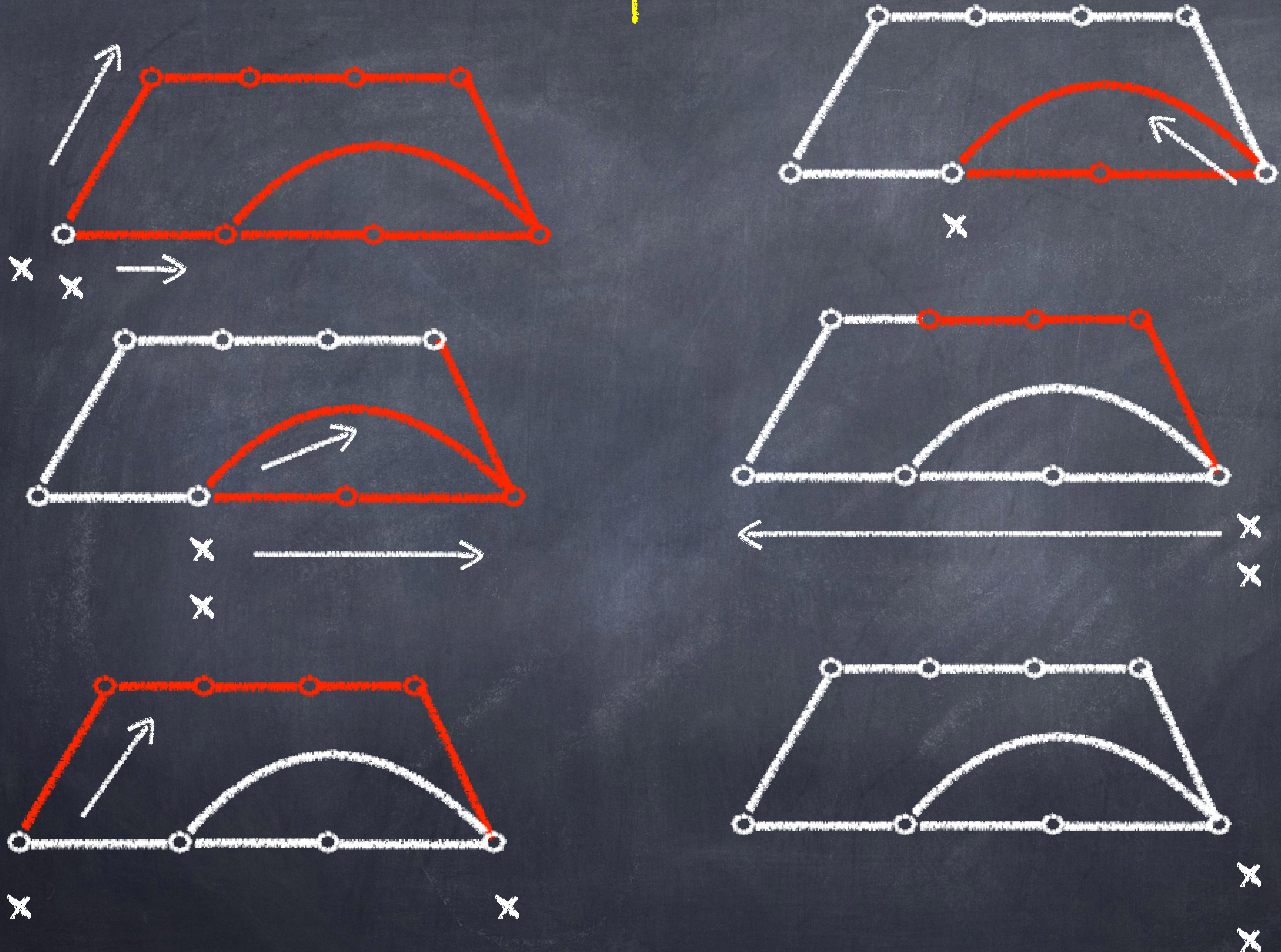
“Simple” case: fugitive and searchers move at the same speed

- We speak about velocity - distances become important
- What is the dynamics of the players?

"Simple" case: fugitive and searchers move at the same speed

- Graph has edges of unit length
- Discrete steps
- Each searcher: either stay in a vertex or move at full speed from vertex to vertex
- Fugitive can move along edges with speed at most 1

Example



Theorem (FF, Petrov, 1998)

For $n \geq 3$, $s_1(K_{n+1}) = n + 1$

Theorem (FF 1995)

For every tree T, $s_1(T) = s_\infty(T)$

Open questions

- Monotonicity does not help but do we need exponential number of steps?
- What is the complexity? Should be at least NP-hard
- Is there a polynomial time algorithm to decide if 2 players suffice?

Open questions

- What are obstructions for small search number?
- Polynomial time algorithm on simple graphs: outerplanar, block graphs, serial-parallel, grids, etc.?

Related Game: 0-visibility cops

J Comb Optim (2015) 29:541–564
DOI 10.1007/s10878-014-9712-6

Zero-visibility cops and robber and the pathwidth of a graph

Dariusz Dereniowski · Danny Dyer · Ryan M. Tifenbach · Boting Yang

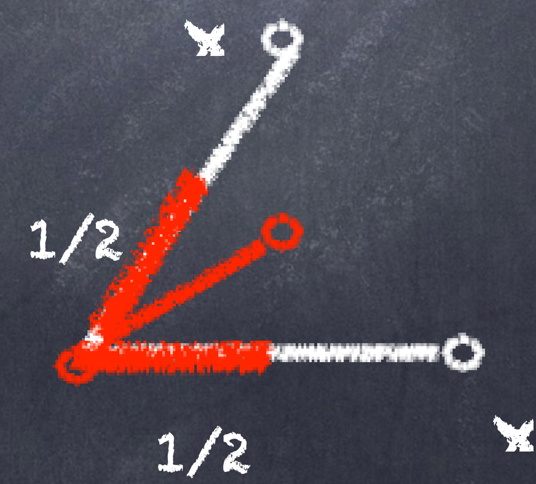
Published online: 19 February 2014

Abstract We examine the zero-visibility cops and robber graph searching model, which differs from the classical cops and robber game in one way: the robber is invisible. We show that this model is not monotonic. We show that the zero-visibility copnumber of a graph is bounded above by its pathwidth and cannot be bounded below by any nontrivial function of the pathwidth. As well, we define a monotonic version of this game and show that the monotonic zero-visibility copnumber can be bounded both above and below by positive multiples of the pathwidth.

When searchers are faster

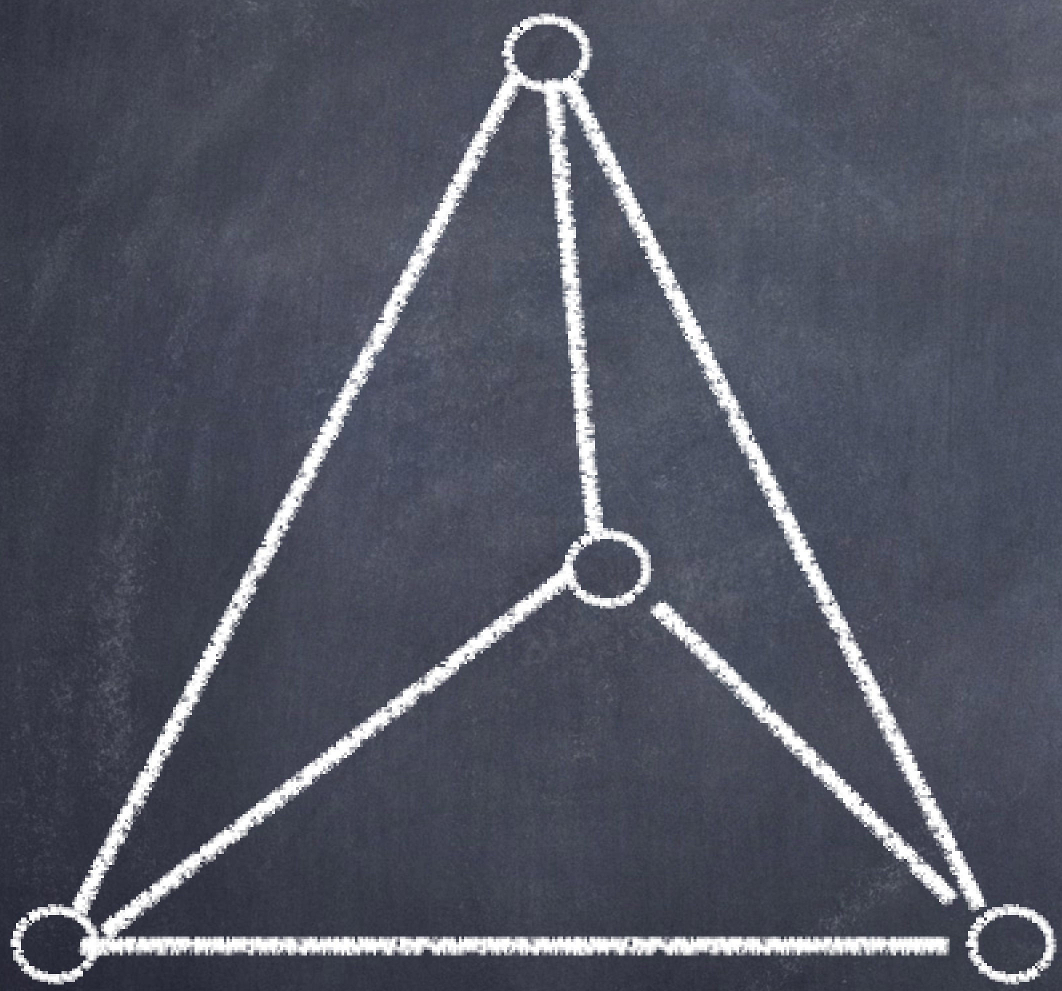
When searchers are faster

2 searchers, p=2



- For p, when searher pass one edge, fugitive pass 1/p of an edge

Very specific question

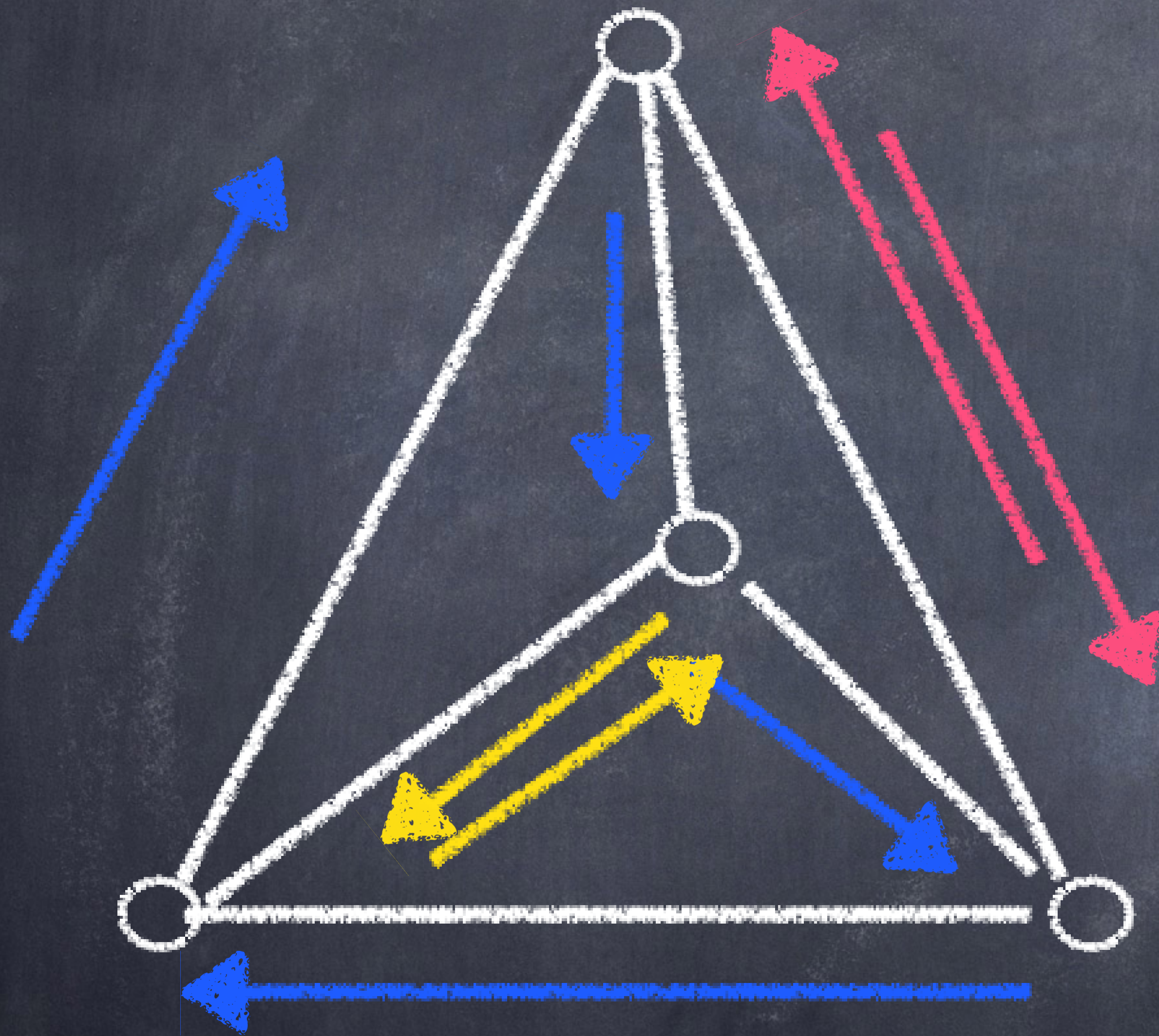


How to search tetrahedron?

4 searchers win with any speed >0

Very specific question

How to search tetrahedron?



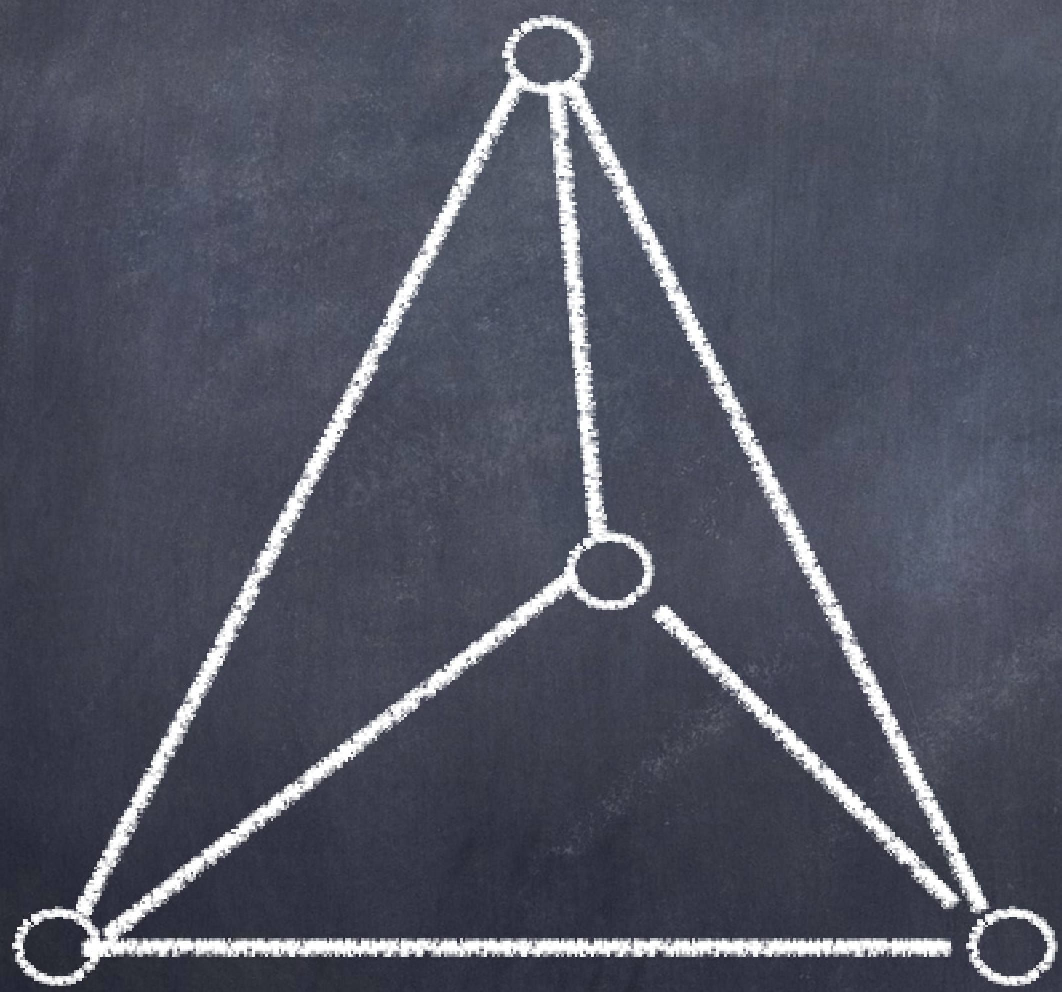
3 searchers will do for

$$p = 1 + \varepsilon$$

For $p=1$, one can show that 3 searchers are not enough

Very specific question

How to search tetrahedron?



2 searchers win for p=3

Very specific open question

For every p compute the minimum number of searchers sufficient to win on a tetrahedron T

Conjecture:

$$Sp(T) = \begin{cases} 4, & \text{for } p \le 1 \\ 3, & \text{for } 1 < p < 3 \\ 2, & \text{for } 3 \le p \end{cases}$$

Algorithmic questions on trees

- Given a tree T and integer p, how many searchers required to succeed on T? Open for $p=2$
- Given a tree T and integer k, what is the minimum p such that k searchers succeed on T? Open for $k=2$
- Given a tree, decide if $k+p<=4$?

Grids

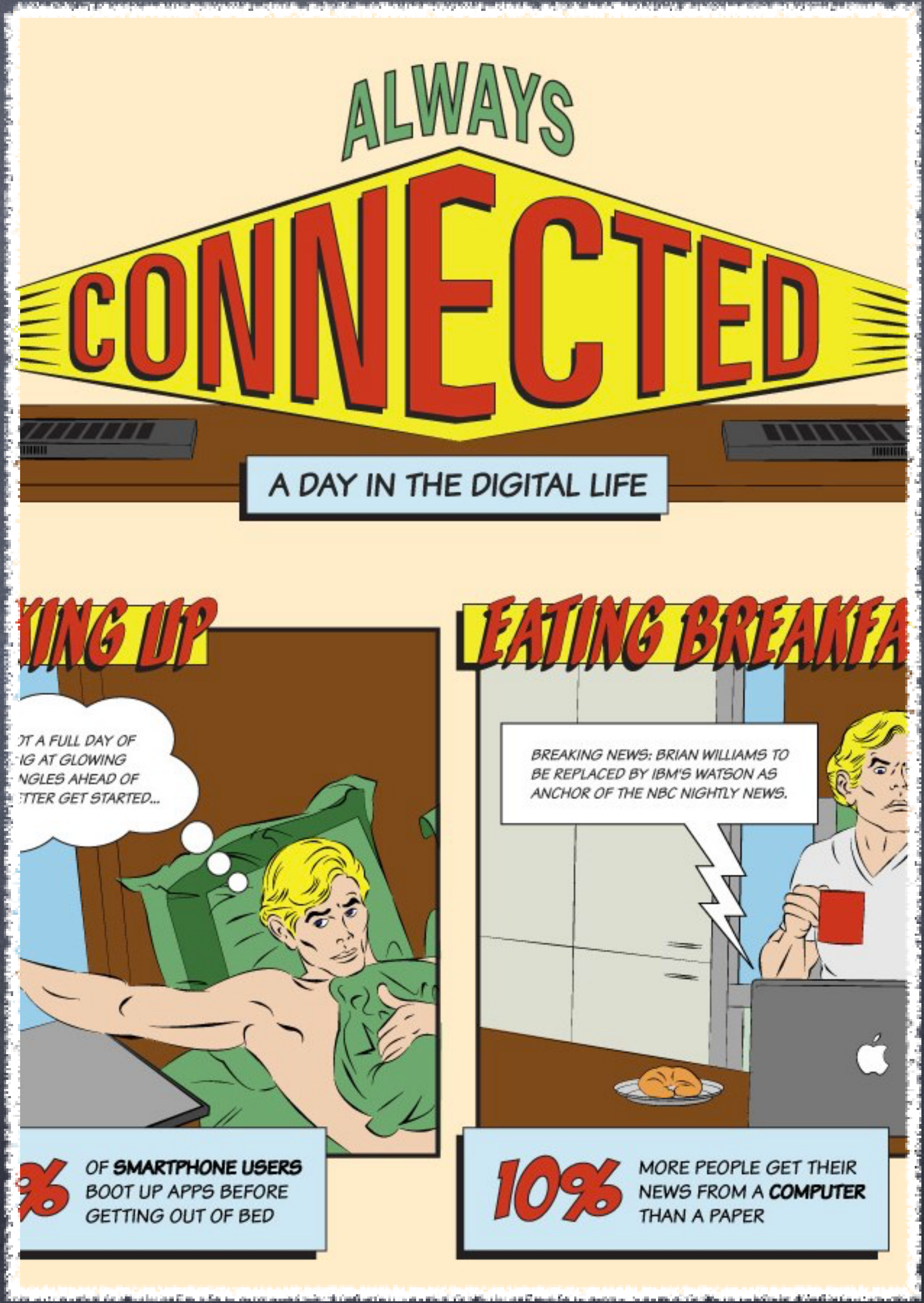
- The search number of a $n \times n$-grid (when fugitive is arbitrarily fast) is $n+1$
- What about the speed restrictive case?
- Does $p=1$ give advantage to fugitives?

Things even more complicated when we think of graphs as of topological structures (back to caves!!!)

Stay connected!

or

Connected Graph Searching



Clearing with constrains

Capture of an Intruder by Mobile Agents

Lali Barrière[*] Paola Flocchini[†] Pierre Fraigniaud[‡] Nicola Santoro[§]

SPAA 2002

ABSTRACT

Consider a team of mobile software agents deployed to capture a (possibly hostile) *intruder* in a network. All agents, including the intruder move along the network links; the intruder could be arbitrarily fast, and aware of the positions of all the agents. The problem is to design the agents' strategy for capturing the intruder. The main efficiency parameter is the size of the team. This is an instance of the well known *graph-searching* problem whose many variants have been extensively studied in the literature. In all existing solutions, and in all the variants of the problem, it is assumed that agents can be removed from their current location and placed in another network site arbitrarily and at any time. As a consequence, the existing optimal strategies cannot be employed in situations for which agents cannot access the network at any point, or cannot "jump" across the network, or cannot reach an arbitrary point of the network via an internal travel through insecure zones. This motivates the *contiguous search problem* in which agents cannot

if T is a processor-network, then the minimal search strategy for T can be computed by T in a decentralized manner, using a linear number of messages.

Keywords: Graph-searching, mobile agent, network intruder.

Categories & Subject Descriptors: G.2.2 and F.2.2.

General Terms: Algorithms.

1. INTRODUCTION

Networked environments which support mobile agents can be penetrated by possibly harmful agents, called *intruders*. Concern for the severe damage intruders can cause has motivated a large amount of research, especially on detection, whose focus is on solutions by teams of mobile agents (e.g., see [1, 13, 15, 16, 31, 37]). Once the presence of an intruder is detected, a team of mobile system agents is deployed to capture it. Both the intruder and the agents move along the network links, but the intruder could be arbitrarily fast,

Connected search is not monotone

Discrete Mathematics 309 (2009) 5770–5780

Contents lists available at ScienceDirect

Discrete Mathematics

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

ELSEVIER

Sweeping graphs with large clique number

Boting Yang[a], Danny Dyer[b,*], Brian Alspach[c]

[a] *Department of Computer Science, University of Regina, Canada*

[b] *Department of Mathematics and Statistics, Memorial University of Newfoundland, Canada*

[c] *Department of Mathematics and Statistics, University of Regina, Canada*

ARTICLE INFO

Article history:
Received 12 June 2006
Accepted 28 May 2008
Available online 2 July 2008

Dedicated to Pavol Hell on the occasion of his sixtieth birthday

Keywords:
Edge searching
Sweeping

ABSTRACT

Searching a network for intruders is an interesting and often difficult problem. Sweeping (or edge searching) is one such search model, in which intruders may exist anywhere along an edge. It was conjectured that graphs exist for which the connected sweep number is strictly less than the monotonic connected sweep number. We prove that this is true, and the difference can be arbitrarily large. We also show that the clique number is a lower bound on the sweep number.

Connected search via normal search
Ratio 2 problem

connected Search (G)/search (G) <=2+o(1)

SIAM J. Discrete Math.
Vol. 26, No. 4, pp. 1709–1732

FROM PATHWIDTH TO CONNECTED PATHWIDTH*

DARIUSZ DERENIOWSKI†

Abstract. It is proven that the connected pathwidth of any graph G is at most $2 \cdot \mathrm{pw}(G) + 1$, where $\mathrm{pw}(G)$ is the pathwidth of G. The method is constructive, i.e., it yields an efficient algorithm that for a given path decomposition of width k computes a connected path decomposition of width at most $2k+1$. The running time of the algorithm is $O(dk^2)$, where d is the number of "bags" in the input path decomposition. The motivation for studying connected path decompositions comes from the connection between the pathwidth and the search number of a graph. One of the advantages of the above bound for connected pathwidth is an inequality $\mathtt{cs}(G) \leq 2\mathtt{s}(G) + 3$, where $\mathtt{cs}(G)$ and $\mathtt{s}(G)$ are the connected search number and the search number of G, respectively. Moreover, the algorithm presented in this work can be used to convert a given search strategy using k searchers into a (monotone) connected one using $2k + 3$ searchers and starting at an arbitrary homebase.

Key words. connected pathwidth, connected searching, fugitive search games, graph searching, pathwidth

AMS subject classifications. 05C83, 68R10, 05C85

Algorithmic questions about connected search

- The problem can be shown to be NP-hard. Is it in NP?
- Do we ever need programs with exponential number of steps?

Algorithmic questions about connected search

- For fixed number of searchers k, is the problem in P?

Algorithmic questions about connected search

- Is connected search number FPT parameterized by k?
- On trees in P, weighted trees NP-hard
- Is it in P on outerplanar graphs?
- Graphs of bounded treewidth?

Other constrains?

- Connected search: the diameter of the cleared subgraph is always finite. What if we want to optimize the diameter? Steiner tree between searchers?
- How this is related to normal search? Monotonicity? Complexity? Approximation?

Information counts

Information

- We know where fugitive is at every step (treewidth)
- We do not know where fugitive is (pathwidth)
- What if we have option for q queries to an oracle to ask about position of the fugitive?

This variant of graph searching is monotone!

Available online at www.sciencedirect.com

ScienceDirect

Theoretical Computer Science 399 (2008) 169–178

Theoretical Computer Science

www.elsevier.com/locate/tcs

Monotonicity of non-deterministic graph searching

Frédéric Mazoit[a], Nicolas Nisse[b,*]

[a] *LABRI, University of Bordeaux, 33405 Talence, France*
[b] *LRI, University of Paris Sud, 91405 Orsay, France*

Abstract

In graph searching, a team of searchers are aiming at capturing a fugitive moving in a graph. In the initial variant, called *invisible graph searching*, the searchers do not know the position of the fugitive until they catch it. In another variant, the searchers permanently know the position of the fugitive, i.e. the fugitive is visible. This latter variant is called *visible graph searching*. A

Algorithms for non-deterministic search

Complexity on trees for fixed q

Theoretical Computer Science 580 (2015) 101–121

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Non-deterministic graph searching in trees ☆

Omid Amini [a], David Coudert [b,c,*], Nicolas Nisse [b,c]

[a] CNRS – DMA, École Normale Supérieure, Paris, France
[b] Inria, France
[c] Univ. Nice Sophia Antipolis, CNRS, I3S, UMR 7271, 06900 Sophia Antipolis, France

A R T I C L E I N F O

Article history:
Received 21 October 2013
Received in revised form 12 August 2014
Accepted 24 February 2015

A B S T R A C T

Non-deterministic graph searching was introduced by Fomin et al. to provide a unified approach for pathwidth, treewidth, and their interpretations in terms of graph searching games. Given $q \geq 0$, the q-limited search number, $s_q(G)$, of a graph G is the smallest

Open questions

- The problem is minor-closed, using Robertson-Seymour machinery, there is $f(k,q)\mathrm{poly}(n)$-time algorithm solving it. What is that algorithm?
- For fixed k, what is the minimum number of queries required to search the graph? For $k=2$?
- Pathwidth and treewidth can be found in time $(2-\mathrm{eps})^n$. What about nondetermenistic search?

Open questions

- For $q>1$, is there a polynomial time algorithm computing the minimum number of searchers with q queries required to search a tree?
- For fixed number of searchers, what is the minimum number of queries to search a tree? (For 2 searchers Amini et al. gave linear time algorithm.)

What of fugitive is not that peaceful?

Information can be different ...

In Game Theory and Applications, Volume X
Edited by L.A. Petrosjan and V.V. Mazalov, pp.1–12
ISBN 1-59454-224-4

GRAPH SEARCHING PROBLEMS WITH THE COUNTERACTION

V.Y. Andrianov
Nikolay N. Petrov
Faculty of Mathematics and Mechanics
St. Petersburg State University
St. Petersburg
Russia

1. Introduction

Graph searching problems have attracted the attention of many researchers because of numerous (sometimes unexpected) applications in various fields of Discrete Mathematics. These problems arise in some computer-virus protection models ([1]) and in the VLSI theory where many important parameters of graph layout such as the cutwidth ([2]), the bandwidth ([3]), the vertex separation number ([4]) have the search-theoretic interpretation. There are important connections of graph search-

Chess model: Invisible king

- Velocities of players are equal
- 0-visibility searchers
- fugitive should move at every step
- fugitive can eat searchers

Checkers model:

- Velocities of players are equal
- 0-visibility searchers
- fugitive should move at every step
- fugitive must eat searchers

Questions:

- Formula are known for paths and cycles
- No complexity is known
- Algorithm for trees?
- Poly-algorithm for a fixed number (=2) of searchers?
- What if there are several fugitives?
- Dual problem: How many aggressive fugitives do we need to win the game?

Conclusion

- In 2017: 50 years of Graph Searching
- We went very far but...
- with very restricted model
- Still long path to go?...