J. Parallel Distrib. Comput. 88 (2016) 1–15

Contents lists available at ScienceDirect

J. Parallel Distrib. Comput.

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

Black hole search in computer networks: State-of-the-art, challenges and future directions

Mengfei Peng[a], Wei Shi[a,*], Jean-Pierre Corriveau[b], Richard Pazzi[a], Yang Wang[c]

[a] Faculty of Business and Information Technology, University of Ontario Institute of Technology, Oshawa, Canada
[b] School of Computer Science, Carleton University, Ottawa, Canada
[c] Shenzhen Institute of Advanced Technology, Chinese Academy of Science, Shenzhen, China

HIGHLIGHTS

- We survey 35 single BHS, 9 multiple BHS papers, 14 papers on other malicious hosts.
- We use four categories to classify all surveyed papers.
- We compare papers by their complexity and introduce typical techniques.
- We analyse the impact of different assumptions and point out key observations.
- We mention research areas that are related to BHS and point out future directions.

ARTICLE INFO

Article history:
Received 27 June 2014
Received in revised form
22 June 2015
Accepted 21 October 2015
Available online 31 October 2015

Keywords:
Black hole search
Mobile agents
Multiple black holes
Malicious host
Network diagnosis

ABSTRACT

As the size and use of networks continue to increase, network anomalies and faults are commonplace. Consequently, effective detection of such network issues is crucial for the deployment and use of network-based services. In this paper, we focus on one specific severe and pervasive network problem, namely the presence of one or more *black holes*. A black hole models a network node that is accidentally off-line or in which a process deletes any visiting agent or incoming data upon arrival without leaving any observable trace. *Black Hole Search* is the process that leverages mobile agents to locate black holes in a fully distributed way. In this paper, we review the state-of-the-art research in this area. We first distinguish between solutions for synchronous and asynchronous networks. We then consider the communication model between agents, their starting locations and the topological knowledge each may hold. We also report on the proposed algorithms with respect to their complexity and correctness. We remark that most existing work addresses locating a single black hole, multiple black hole search being significantly more complex. We not only summarize major results in this area but also briefly touch on other types of malicious hosts. Finally, we identify some open problems for future research.

1. Introduction

Over the past few decades, as network-based services have become prevalent, so has the need for effective diagnosis of all-too-frequent network anomalies and faults. Among these, a *black hole* is a severe and pervasive problem. A black hole models a computer that is accidentally off-line or a network site in which a resident process (e.g., an unknowingly-installed virus) deletes any visiting agents or incoming data upon their arrival without leaving any observable trace [30]. For example, in a cloud, a node that causes loss of essential data (for the system and/or its users) constitutes a black hole and *de facto* compromises the quality of any service in this cloud. Similarly, any undetectable crash failure of a site in a network transforms that site into a black hole.

A *mobile agent* is an abstract and autonomous software entity. As such, agents are versatile and robust in changing environments, and can be programmed to work in cooperative teams. Members of such teams may have different complementary specialities, or be duplicates of one another [53]. For *black hole search*, one or a team of identical agents are generally used. These agents have limited computing capabilities and bounded storage. They all obey an identical set of behavioural rules (referred to as the "protocol")

[*] Corresponding author.
E-mail addresses: mengfei.peng@uoit.ca (M. Peng), wei.shi@uoit.ca (W. Shi), jeanpier@scs.carleton.ca (J.-P. Corriveau), richard.pazzi@uoit.ca (R. Pazzi), yang.wang1@siat.ac.cn (Y. Wang).

http://dx.doi.org/10.1016/j.jpdc.2015.10.006
0743-7315/© 2015 Elsevier Inc. All rights reserved.

and can move from a node to a neighbouring one. Also, these agents are anonymous (i.e., do not have distinct identifiers) and autonomous (i.e., each does its own computing and uses its own memory).

Using such agents offers several potential advantages: they can reduce network load, overcome network latency, encapsulate protocols, execute asynchronously and autonomously, and even adapt dynamically [61]. For example, black hole search may instead rely on the use of a central controller. In this case, the latter must constantly send Ping messages to nodes or, alternatively, require that each node send it periodically a message confirming this node's activity. Both of these strategies lead to heavy network traffic that can be avoided when using mobile agents for such a search.

Consequently, in this paper, black hole search is scoped to be a task that allows a team of mobile agents to collaborate with each other to locate black holes within finite time while eventually leaving at least one agent to survive and know all the edges leading to black holes [37]. (We abstract a network into a graph $G(V, E)$ where nodes in V represent computer hosts and edges in E represent network links.) Currently, many distinct approaches to using mobile agents to locate a *single* black hole in a computer network have been studied in many different contexts (e.g., [6,17,19,28,43,57]). Generally, existing solutions rest on anonymous agents that all execute the same protocol to identify and report any black hole.

In 2006, Flocchini et al. [50] scrutinized the black hole search problem for both asynchronous and synchronous networks. That survey also introduced the black hole search problem as a special case of exploring and mapping an unknown environment. While there exists a large body of literature on unknown graph exploration problems, it is mostly irrelevant to this paper for it generally assumes that the underlying network graph does not contain any type of malicious entities [2]. Conversely, work on *dangerous graph search* (e.g., [18]) does address the detection and localization of malicious hosts (such as black holes), malicious agents, and faulty links. In particular, in their 2012 survey [64], Markou et al. discussed previous research on identifying hostile nodes. They mainly focused on synchronous special trees, arbitrary trees and arbitrary graphs, with a brief mention of asynchronous rings. More recently, Zarrad et al. [69] briefly discuss solutions for black hole search in synchronous and asynchronous networks, however without analysing the underlying assumptions of these solutions.

In this paper, our goal is to review the state-of-the-art in the black hole search field in order to help readers understand the existing work, as well as grasp some of the remaining challenges in this field. We specifically exclude from the scope of this paper the issue of *black hole attacks* [3,8,68], which is superficially related to the topic at hand.

First, we introduce the main models and assumptions that are commonly used in the relevant literature with respect to network synchronization, the communication model between agents, their starting locations and the topological knowledge each may hold. In addition to obtaining models, determining their complexity is also critical for the actual deployment of the proposed algorithms. Possibly relevant measures of complexity include the total number of moves, the number of agents, the number of tokens, and the memory footprint, as well as algorithm efficiency *per se*. Time cost is another metric that is usually discussed when considering synchronous networks. Because the time cost of transit (i.e., moving from one node to a neighbouring one) is unpredictable in asynchronous networks, in such networks time complexity can only be measured using additional assumptions such as: it takes an agent an unitary amount of time (i.e., one 'time unit') to traverse a link or explore a node (which amounts to having a global clock) [5,6,30].

We then separate the papers of this survey based on their network synchronization (i.e., synchronous or asynchronous). The motivation for this is simple: when considering synchronization, the black hole search problem is very different with respect to it allowed behaviour(s), its inherent difficulty and its limitations, and so are the proposed solutions. For both categories, we further classify the studies based on the agent communication model, the agent starting locations, and knowledge of the network. That is, we contrast the proposed solutions with respect to their choice of assumptions (and resulting complexity) in each of the three areas of variability just mentioned for black hole search. Beyond such comparisons, we also briefly introduce some open problems that persist in this field.

More specifically, the rest of this paper is organized as follows: Frequent assumptions and models for black hole search are introduced in Section 2, then relevant measures of complexity are discussed in Section 3. Solutions for the detection of a single black hole in synchronous and asynchronous networks are respectively addressed in Sections 4 and 5. In Section 6 we consider multiple black holes search. We then report in Section 7 on the most recent results pertaining to the different types of malicious hosts. In Section 8, we summarize the contributions we survey and mention some open problems stemming from this work. We draw some conclusions in Section 9.

2. Common models and assumptions

Because none of the existing algorithms are able to solve the black hole search problem without some restrictions, it is crucial to gather the assumptions that are typically made in existing research and study the impact of each one. In this section, we introduce a list of such assumptions.

To start with, existing work always assumes that the agents' initial wake-up nodes are safe. Otherwise, all the agents may die before even starting graph exploration, rendering the problem unsolvable. Furthermore, unless the agents are extremely fortunate, (viz., happen to explore all nodes in a graph except the black hole(s)) in order to systematically identify a black hole, we must expect at least one agent to go in a black hole and somehow leave a hint for the other agents before it dies, which eventually allows the surviving agents to know the location of the black hole(s). All other common assumptions are listed in Table 1. We will now provide a detailed explanation of each of these assumptions.

2.1. Network synchronization

2.1.1. Synchronous network

A *synchronous network* is a network in which all agents initially wake up at the same time and where it takes a quantum amount of time (called a *time unit*) for an agent to traverse a link or explore a node: All agents are thus synchronized with respect to a global clock. By the end of each time unit, an agent must decide whether to move to a neighbouring node, or stay at its current node, or terminate the algorithm. As such, the complexity of the agent's algorithm in synchronous networks can be measured in terms of the number of time units.

In synchronous networks, a *time-out mechanism* is available to enforce the time synchronization [17,22–24,56]. Such a mechanism allows us to easily identify which agents died in the black hole(s). Suppose a team of agents should meet at a node u after m time units, after this time-out, all other agents know that those that do not show up in node u died in the black hole(s).

Using such a time-out mechanism, the black hole can be located using only 2 agents in any network that has only one black hole present when a network map is available for every agent. In this case the network size is not required to guarantee a solution. For