

Chapter 66

An Artificial Bee Colony Algorithm for Detecting Distributed Predicates

Eslam Al Maghayreh

King Saud University, Riyadh, Saudi Arabia

ABSTRACT

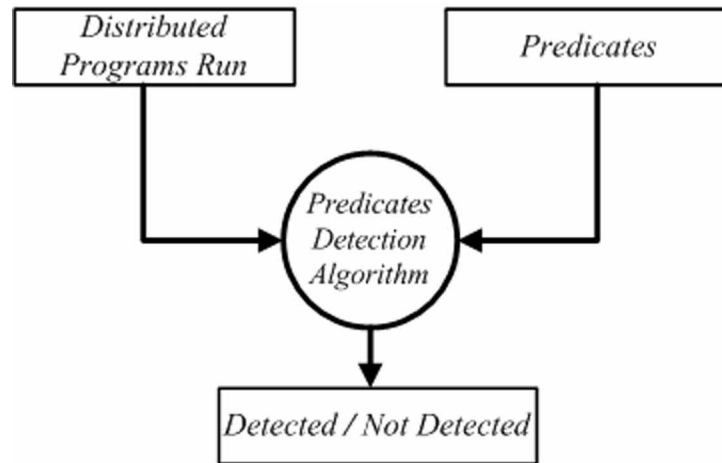
Developing correct dependable distributed applications is a difficult task. The techniques developed to detect distributed predicates can be used to verify the satisfaction of specific properties (represented as predicates) in a given run of a distributed application. These techniques have several applications including the verification of the correctness of a particular implementation of a distributed system (runtime verification). Due to concurrency, the number of states to be considered in detecting a distributed predicate is exponential. This article investigates the effectiveness of using a meta-heuristic algorithm in solving the problem of detecting distributed predicates. Namely, the authors have used the Artificial Bee Colony algorithm to develop an algorithm to detect distributed predicates. The conducted experiments demonstrate that the ABC-based detection algorithm has a better performance compared with the traditional state enumeration-based detection algorithm.

INTRODUCTION

The problem of detecting distributed predicates (runtime verification) has attracted attention in the domain of asynchronous distributed systems (Yingchareonthawornchai et al., 2017; Natarajan et al., 2017; Chang and Garg, 2016; Zhu et al., 2016; Chauhan et al., 2013; Freiling and Jhumka, 2007; Garg, 2002; Dumais and Li, 2002). Several researchers have developed techniques to solve this problem. These techniques can be used to verify the satisfaction of some predefined properties (represented as predicates) in a run of a distributed application. Figure 1 demonstrates the distributed predicates detection environment (Maghayreh, 2010).

DOI: 10.4018/978-1-7998-5339-8.ch066

Figure 1. Distributed predicates detection environment (Maghayreh, 2010)



Distributed predicates detection techniques operate on a specific implementation of a distributed system. On the other hand, model checking techniques operate on an abstract model of the system. As a result, distributed predicates detection techniques can be more powerful in reasoning about the correctness of a specific implementation of a distributed system (Garg, 2002; Maghayreh, 2010).

Developers of distributed applications can use detection techniques in testing and debugging. For example, to implement one of the most common debugging commands “Stop the program when predicate P becomes true”, you need to detect the satisfaction of P . If predicate P involves variables of different processes in a distributed program, then the development of an efficient detection algorithm might be a very difficult task.

Due to concurrency, detecting distributed predicates is a very difficult and time-consuming job. Researchers have suggested several techniques to decrease the cost of detecting certain classes of distributed predicates (Zhu et al., 2016; Chauhan et al., 2013; Maghayreh, 2010; Garg, 2002). In Section 3, we will summarize some of these techniques. However, it has been proved that detecting a distributed predicate in a run of a distributed application is an NP-complete problem in general (Garg, 2002).

In (Karaboga and Basturk, 2008; Karaboga and Akay, 2009), a swarm intelligent algorithm known as the artificial bee colony (ABC) algorithm has been proposed. The ABC algorithm simulates the behavior of real bees in searching for food sources and exchanging the information with other bees. The main idea of the ABC algorithm is not complex, and it's easy to implement. The ABC algorithm has been exploited in solving several problems in different domains (Senberber and Bagis, 2017; Koza and Karaboga, 2017; Ozdemir et al., 2017; Shaik and Reddy, 2016; Yang et al., 2015). In this paper, we have utilized the ABC algorithm in developing a powerful distributed predicates detection technique. The proposed technique can detect the satisfaction of any distributed predicate under the possibly modality (the possibly modality will be defined in the next section). Several experiments have been conducted to evaluate the performance of the ABC-based detection technique.

The rest of this paper is structured as follows. Section 2 presents a formal model of a distributed application's run. We will discuss other related works in Section 3. A brief description of the ABC algorithm will be presented in Section 4. We will explain the details of the developed ABC-based detection

11 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/an-artificial-bee-colony-algorithm-for-detecting-distributed-predicates/275343

Related Content

Roof to Technology Implementation: The Adoption of Cloud Concept in Various Areas

Inderbir Kaur (2021). *Research Anthology on Architectures, Frameworks, and Integration Strategies for Distributed and Cloud Computing* (pp. 1602-1614).

www.irma-international.org/chapter/roof-to-technology-implementation/275356

Unique Taxonomy for Evaluating Fog Computing Services

Akashdeep Bhardwaj and Sam Goundar (2021). *Research Anthology on Architectures, Frameworks, and Integration Strategies for Distributed and Cloud Computing* (pp. 165-177).

www.irma-international.org/chapter/unique-taxonomy-for-evaluating-fog-computing-services/275284

Distributed Computing for Internet of Things (IoT)

Dharmendra Trikamlal Patel (2021). *Research Anthology on Architectures, Frameworks, and Integration Strategies for Distributed and Cloud Computing* (pp. 1874-1894).

www.irma-international.org/chapter/distributed-computing-for-internet-of-things-iot/275369

Efficient Fault Tolerance on Cloud Environments

Sam Goundar and Akashdeep Bhardwaj (2021). *Research Anthology on Architectures, Frameworks, and Integration Strategies for Distributed and Cloud Computing* (pp. 1231-1243).

www.irma-international.org/chapter/efficient-fault-tolerance-on-cloud-environments/275336

Building a Secured XML Real-Time Interactive Data Exchange Architecture

Yousef E. Rabadi and Joan Lu (2021). *Research Anthology on Architectures, Frameworks, and Integration Strategies for Distributed and Cloud Computing* (pp. 270-357).

www.irma-international.org/chapter/building-a-secured-xml-real-time-interactive-data-exchange-architecture/275289