Chapter 27
A Multi-Agent-Based Approach for Critical Components Identification and Testing

D. Jeya Mala
Thiagarajar College of Engineering, India

R. Iswarya
Thiagarajar College of Engineering, India

ABSTRACT

In real time software systems, testing plays a crucial role as any of the critical components in these systems are left undetected, then inadvertent effects will happen which will lead to erroneous operations, system failure, high cost and resource wastage etc. To address this most important and the emergent problem, this research work proposes an effective method by means of multi-agents based approach to identify such critical components and execute test cases along the critical test paths which will aid in effectively covering them during testing. Finally, this paper also compared the performance with existing approaches in terms of time taken for the search process and the component coverage based test adequacy criterion to ensure quality of the software.

1. INTRODUCTION

As per the study of National Institute of Standards and Technology, the cost for an inadequate infrastructure for software testing is estimated to be from \$22.2 to \$59.5 billion (Tassey, 2002). As exhaustive testing (testing 100%) is not feasible (David et al. 2005, Myers, 1979), the industries are forced to stop the testing process at one point of time and deliver the software to the customers. This leads them to compromise the quality of the software due to customers' need for quick delivery of quality software, reduced software development lifecycle, changing markets with global competition and rapid development of new processes and technologies.

DOI: 10.4018/978-1-5225-3923-0.ch027

The surveys have indicated that, many of the complex systems' failures are due to insufficient testing of software before they are deployed to the customer side (Bernardi, 2011; Schneidewind, 1978). After the analysis, it has been identified that, the highly critical components are not being properly tested or simply ignored without knowing their critical level due to time and cost compromises.

The identification of such critical components from the software prior to testing is still a research area, since the automated testing tools available in the market doesn't address the said problem. Based on our field surveys conducted in several organizations during the past months, it has been identified that, most of the defects reported by the customers after delivery are present in these higher critical components. This gives us the insight on the importance of critical components identification and their verification prior to the delivery of the software. But, the identification of the criticality level of a component involves the evaluation of various metrics and measures associated with them.

Hence, an automated software testing framework that can identify and prioritize the critical components based on the various metrics and measures associated with each of the components and can also provide an optimized critical paths list which can reduce the time and cost needed in the testing process without compromising the testing of these critical components is the need of the hour.

From the literature survey, it has been identified that only a very few works have been conducted in the said research area, and that too have been limited by the type and number of metrics used by them. If the software under test is small and simple then the identification can be done manually. As the real time complex systems have huge functionalities, there is a need for an approach that embeds both intelligence and automation as a tool.

Since software testing is NP-hard (Non-Polynomial hard) (Nagappan, 2006), and as manual testing is costly and error prone, several existing research works on structural testing have employed computationally intelligent techniques, such as artificial intelligence and evolutionary computation methods, to achieve optimization in the testing process (Alok Singh, 2009, Basturk & Karaboga, 2006, Baykasolu et al.,2009, Dorigo et al., 1996, Teodorović & Dell, 2006, Fathian et al., 2007, Karaboga, 2008, 2009, Pham et al., 2006, Srinivasa Rao et al., 2008, Li-Pei et al., 2008).

As several existing knowledge based approaches are population based approaches with sequential execution behavior, from the literature survey it has been identified that, agents based approach gives promising results when compared to the other solutions. However, this research work applied the multi agents based approach in which each agent is responsible for doing its activities in a parallel manner.

Hence, this research work proposed, an automated critical components identification and verification framework using multi-agents based approach to identify the critical components and to generate and optimize the number of critical paths needed for their verification.

1.1. Problem Formulation

The proposed research problem has its origin from various industrial surveys taken during the recent years. Even though achieving zero-defect quality software is the ambition, it is not possible in reality (Pressman, 2007). Software testing can be very costly and it typically consumes at least 50% of the total cost involved in software development (Bernardi et al., 2011, Li & Lam, 2004). There should be some stopping criterion to be followed to stop testing at one point of time and release the software to