한국컴퓨터정보학회 동계학술대회 논문집 제27권 제1호 (2019. 1)

항공기 소프트웨어의 원자성위배 자율수리 도구를 위한 안전한 인터리빙 정보를 생성하는 기법

백형진[O], 최으뜸[*], 이건표[*], 전용기[*]
[O*]경상대학교 정보과학과
e-mail: {azzp45, slateblue33, gnurvy2, jun}@gnu.ac.kr[O*]

A Technique to Generate Information of Safe Interleavings for On-the-fly Atomicity Violation Repairing in Airborne Software

Hyoung-Jin Baek[O], Eu-Teum Choi[*], Keon-Pyo Lee[*], Yong-Kee Jun[*]
[O*]Dept. of Informatice, Gyeong-Sang University

● 요 약 ●

본 논문은 멀티스레드를 지원하는 항공기 소프트웨어에 적용될 수 있는 자율수리 도구의 문제의 해결방법을 제안하는 논문이다. 기존의 연구는 프로그램의 반복수행을 통해 안전한 인터리빙을 수집하여 프로그램의 동작을 제한한다. 하지만 테스트 단계에서 수집되지 않은 안전한 인터리빙을 잘못된 인터리빙으로 처리하여 수리를 수행함으로써 불필요한 오버헤드가 발생한다. 본 논문은 원자성위배 패턴을 사용하여 안전한 인터리빙을 예측하여 생성시키는 기법을 사용하여 수리기법에서 불필요한 수리로 인한 오버헤드를 감소하기 위한 안전한 인터리빙 정보를 생성하는 기법을 제안한다.

키워드: 항공기 소프트웨어(airborne software), 자율수리(on-the-fly repairing), 동시성오류(concurrency error)

I. Introduction

항공기 소프트웨어는 고신뢰성을 요구하는 임베디드 소프트웨어이다[1]. 동시성오류는 비결정적인 수행결과를 발생시킨다. 이러한 동작은 항공기 소프트웨어의 신뢰성을 보장하기 어렵게 만든다. 따라서 동시성오류는 개발 중 탐지되거나 운행 중 자율적으로 수리가 되어야 한다.

테스트 인터리빙을 사용하여 자율수리를 하는 도구는 프로그램의 반복수행을 통해 안전한 인터리빙을 수집하여 해당 정보를 기반으로 프로그램을 동작하도록 제한한다[2]. 하지만 프로그램 인터리빙 수집 과정 중 수행되지 않은 안전한 인터리빙 또한 잘못된 수행으로 처리하여 제한하기 때문에 불필요한 수리(false positive)를 발생시킨다.

본 논문은 원자성위배가 일어나는 인터리빙 패턴 정보를 사용해 안전한 인터리빙을 예측하여 생성시키는 기법으로 불필요한 수리를 감소시키는 방법을 제안한다.

II. Background

멀티스레드를 사용하는 항공기 소프트웨어에서 동시성오류의 발생은 의도하지 않은 결과를 발생시킬 수 있어 항공기 소프트웨어의 신뢰성을 보장하기 어렵게 만든다. 이러한 동시성오류는 순서위배(order violations), 원자성위배(atomicity violations)가 대표적이다[3]. 순서위배는 서로 다른 스레드들이 공유자원에 접근하는 이벤트의 순서를 보장할 수 없는 것이다. 원자성위배는 원자성이 보장되어야 하는 코드가 개발자의 잘못된 추정으로 인해 원자성이 보장되지 않음을 의미한다. 이러한 동시성오류는 소프트웨어의 실패(failure)를 발생시켜 사고로 이어질 수 있다. 동시성오류 중 원자성위배는 60% 이상을 차지하고 있어 반드시 개발 중에 탐지하여 제거하거나 운행 중에 자율적으로 수리가 되어야 한다[2].

동시성오류의 자율수리는 프로그램을 수행 중에 오류가 발생하였을 때 실패로 이어지지 않고 프로그램을 정상적으로 실행되도록 하는 기법이다. 자율수리 기법 중 테스트 인터리빙 정보를 사용하는 수리기법은 테스트 단계에서 안전한 인터리빙 정보들을 수집한다. 테스트 단계는 프로그램의 반복수행을 통해 안전한 인터리빙을 수집하는 단계이다. 그리고 수집된 안전한 인터리빙 정보를 사용하여 안전한 인터리빙에 속해있지 않으면 동시성오류로 판단하여 수리를 수행한다. 하지만 테스트 단계에서 수행되지 않은 안전한 인터리빙을 잘못된 인터리빙으로 판단하여 수리를 수행하기 때문에 불필요한 오버헤드가 발생한다.

III. Safe Interleaving Generator

본 논문은 항공기 소프트웨어에서 발생할 수 있는 원자성위배 패턴을 이용하여 안전한 인터리빙을 예측하는 기법제시 한다. 그

결과 불필요한 수리를 감소하여 오버헤드가 줄어드는 기법을 제시한다. Table 1은 기법의 예시를 보여준다.

Table 1. anticipated safe interleaving using atomicity violation patterns

원자성위배 패턴	예측된 정상 실행	
R1-W2-W1	R1-W1-W2	W2-R1-W1
W1-R2-W1	W1-W1-R2	R2-W1-W1
R1-W2-R1	R1-R1-W2	W2-R1-R1

Table 1에서 원자성위배 패턴은 공유메모리에 접근 순서가 원자성 위배를 발생시키는 것이다. 하나의 스레드1 에서 원자성이 보장되어야 하는 실행이 스레드2의 접근 사건으로 인하여 원자성이 보장되지 않는 것이다. 예측된 정상 실행은 원자성위배를 제외한 실행이다. R/W는 읽기/쓰기 사건을 의미하고 읽기/쓰기 사건 옆의 숫자는 스레드 번호를 의미한다.

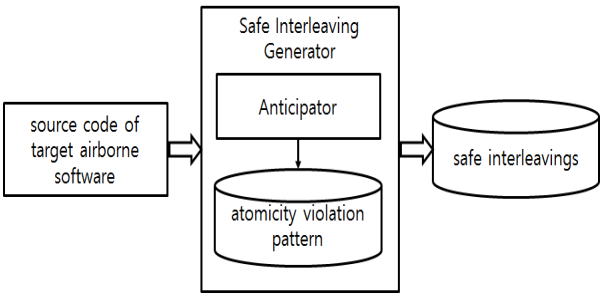

Fig. 1. safe interleaving generator overall architecture

Figure 1은 안전한 인터리빙 생성기의 전체 동작을 나타낸 그림이다. 전체적인 구성은 항공기 소프트웨어 소스코드, 안전한 인터리빙 생성기, 안전한 인터리빙들로 구성이 된다. 안전한 인터리빙 생성기는 항공기 소프트웨어의 소스코드를 입력으로 받는다. 그리고 입력받은 항공기 소프트웨어에서 발생된 원자성위배 패턴을 안전한 인터리빙 생성기에서 사용한다. 예측기는 원자성위배 패턴을 입력받아 예측된 정상 실행들을 만들게 된다. 따라서 안전한 인터리빙 생성기를 사용하면 원자성위배 패턴만을 입력받아 예측된 정상 실행을 알 수 있으므로 테스트 단계에서 안전한 인터리빙이 발생하지 않더라도 안전한 인터리빙 패턴을 예측할 수 있어 자율수리 기법의 불필요한 수리를 방지할 수 있다.

IV. Conclusions

본 논문은 항공기 소프트웨어에서 안전한 인터리빙을 기반으로 동시성오류를 수리하는 기법에서 발생되는 불필요한 수리의 발생을 방지하는 기법을 제안한다. 안전한 인터리빙 생성기법 적용한 후 안전한 인터리빙을 예측하여 자율수리 도구의 불필요한 수리를 감소시켜 오버헤드를 줄이는 기법을 제안한다.

향후 안전한 인터리빙을 예측하는 기법에 관한 연구를 진행할 예정이다. 또한, 실제 수리도구에 적용하여 실험을 진행할 것이다.

ACKNOWLEDGEMENT

이 논문은 2018년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(NRF-2018R1D1A3B07041838)

REFERENCES

[1] O. K. Ha, G. M. Tchamgoue, J. B. Suh, and Y. K. Jun, “On-the-fly healing of race conditions in ARINC-653 flight software,” IEEE, Digital Avionics Systems Conference (DASC), 2010 IEEE/AIAA 29th, pp. 5-A, 2010, October.

[2] M. Zhang, Y. Wu, S. Lu, S. Qi, J. Ren, and W. Zheng, “A lightweight system for detecting and tolerating concurrency bugs,” IEEE Transactions on Software Engineering, 10, 899-917, 2016.

[3] S. Lu, S. Park, E. Seo, and Y. Zhou, “Learning from mistakes: a comprehensive study on real world concurrency bug characteristics,” ACM SIGOPS Operating Systems Review, 42(2), 329-339, 2008.