Canh Minh Do

4-103 Building 4, JAIST Student Housing
1-8 Asahidai, Nomi, Ishikawa 923-1211 Japan
https://canhminhdo.github.io

Phone: (+84) 973-947-403
Email: canhdominh@jaist.ac.jp
Alt: minhcanh99@gmail.com

PARTICULARS

EDUCATION

Japan Advanced Institute of Science and Technology — JAIST, Japan
Ph. D. in Computer Science — *October 2019-*

Japan Advanced Institute of Science and Technology — JAIST, Japan
M. S. in Computer Science — *September 2019*

National Economics University — NEU, Vietnam
BS. in Computer Science — *September 2013*

CURRENT STATUS

Japan Resident, Citizen of Vietnam.

RESEARCH INTERESTS

My research interests are mainly in the areas of formal methods and software verification. How to formalize concurrent and distributed systems as state machines from which we can specify it in a formal specification language, such as Maude and CafeOBJ. To formally verify that state machines enjoy desired properties, model checking and theorem proving are used. Recently, I have been learning about machine learning and deep learning that could be applied to software verification.

DISSERTATION

Title: "A Scalable Testing Technique for Concurrent Programs"
Advisor: Prof. Kazuhiro Ogata

My current dissertation is about developing a scalable testing technique for testing concurrent Java programs by using Java Pathfinder (JPF). JPF is one of the most mature software model checker, which is developed by NASA, for detecting error lurking in concurrent Java programs. However, the use of JPF often leads to the state space explosion due to the non-determinism of thread interleavings. My technique presents a divide & conquer approach to splitting the entire state space into multiple layers, in which each layer consists of many sub-state spaces so that JPF can independently verify such sub-state spaces in parallel. Besides that, I have been trying to find a good criterion for prioritizing selectively states from which JPF should first tackle to find error lurking in programs quickly.

ACADEMIC HONORS

- Special Doctoral Research Fellow (DRF) Scholarship from JAIST for Ph.D. Program, 09/2019 - 08/2022.
- Scholarship From VNU-JAIST to Join the 1+1 Program from JAIST for Master Program, 09/2018 – 08/2019.
- Certificate of Hanoi Communist Youth Union for Top Graduating Students in Hanoi City, 09/2013.
- Highest GPA Graduate Student 2009-2013 in Computer Science Major at University, 09/2013.

WORK EXPERIENCE

- **Senior Software Engineer, TechAcademy Co ., Ltd**, Mar 2015 - Sep 2018. I have involved many projects that need to build the back-end and front-end parts. The former mainly uses CakePHP and Slim to develop APIs. The latter uses mostly AngularJS and ReactJS. Besides, using Selenium to build semi-automated testing for web applications.

- **Software Engineer, Dotoh Vietnam ., JSC**, Sep 2013 - Jul 2014. Building a GA-like analytic system that can detect which company user comes from when visiting a website. It currently serves more than 100 high traffic websites in Japan. Besides, joining to develop a digital document signing system allows users to request a signature from others for documents such as contracts. Each user must provide a valid certification on behalf of the user when signing documents.

- **Software Engineer, Langmaster ., JSC**, Nov 2012 - Jun 2013. Part of an initial team for developing an in-house CRM web application based on Zend framework that is used to facilitate the business operation at the company.

RESEARCH EXPERIENCE

- **Doctoral student JAIST**, Oct 2019 - Present.
  A Scalable Testing Technique for Concurrent Programs: We have proposed A Divide & Conquer Approach to Testing Concurrent Java Programs with JPF and Maude [1][2]. The proposed technique checks if each execution sequence generated from a concurrent Java program by JPF that can be accepted by a formal specification in Maude on the fly. Some experiments demonstrated that the proposed technique could mitigate the state space explosion. After that, we have also demonstrated that the proposed technique can be used to test concurrent Java programs without using Maude, but JPF. A paper, A Divide & Conquer Approach to Testing Concurrent Programs with JPF [4], has been accepted by APSEC 2020. In this paper, we can quickly detect a state of a Java implementation of the NSPK authentication protocol where the nonce secrecy property is broken. However, we are not able to detect a state in which one-to-many agreement property is broken. This is because the state in which the latter property is broken may be located at a much deeper position. To make it possible to detect a state in which the one-to-many agreement property is broken, we have proposed Parallel Stratified Random Testing for Concurrent Programs [5], where some preliminary experiments are conducted to show some progress toward making it possible to detect such a state. Summary, we have been submitted a journal paper [7] about what we have done so far.
  L+1-Layer Divide & Conquer Approach to Lead-To and Eventually Model Checking: I have been joining to develop a tool that supports L+1-Layer Divide & Conquer Approach to Lead-To and Eventually Model Checking [3][6][8][9]. The tool is developed in Maude on top of Full Maude, which is one possible way to mitigate the state explosion in model checking by splitting the state space into multiple layers specialized to Lead-To and Eventually properties.

- **Master student JAIST**, Oct 2018 - Oct 2019.
  Specification-based testing: For a formal specification S and a concurrent program P, state sequences are generated from P and checked to be accepted by S. We suppose that S is specified in Maude and P is implemented in Java. Java Pathfinder (JPF) and Maude are then used to generate state sequences from P and to check if such state sequences are accepted by S, respectively. Even without checking any property violations with JPF, JPF often encounters the notorious state space explosion while only generating state sequences. Thus, we propose a technique to generate state sequences from P and check if such state sequences are accepted by S in a stratified way [1][2].

PUBLICATIONS

PAPERS

1. <u>Canh Minh Do</u> and Kazuhiro Ogata, "Specification-based Testing with Simulation Relations", *The 31st International Conference on Software Engineering and Knowledge Engineering (31st SEKE)*, 2019.

2. <u>Canh Minh Do</u> and Kazuhiro Ogata, "A Divide & Conquer Approach to Testing Concurrent Java Programs with JPF and Maude", *9th International Workshop, SOFL+MSVL*, 2019.

3. Yati Phyo, <u>Canh Minh Do</u> and Kazuhiro Ogata, "Toward development of a tool supporting a 2-layer divide & conquer approach to leads-to model checking". *International Conference on Advanced Information Technologies (ICAIT)*, 2019.

4. <u>Canh Minh Do</u> and Kazuhiro Ogata, "A Divide & Conquer Approach to Testing Concurrent Programs with JPF", *The 27th Asia-Pacific Software Engineering Conference (APSEC)*, 2020.

5. <u>Canh Minh Do</u> and Kazuhiro Ogata, "Parallel Stratified Random Testing for Concurrent Programs", *20th IEEE International Conference on Software Quality, Reliability, and Security (QRS)*, 2020.

6. Yati Phyo, <u>Canh Minh Do</u> and Kazuhiro Ogata, "A Divide & Conquer Approach to Leads-to Model Checking", *The Computer Journal* , 2020.

7. Moe Nandi Aung, Yati Phyo, <u>Canh Minh Do</u> and Kazuhiro Ogata, "A Divide & Conquer Approach to Eventually Model Checking", *MDPI Journal Mathematics*, 2020.

8. Yati Phyo, <u>Canh Minh Do</u> and Kazuhiro Ogata, "A Support Tool for the L+1-Layer Divide & Conquer Approach to Lead-To Model Checking", *COMPSAC: Intelligent and Resilient Computing for a Collaborative World, 45th Anniversary Conference*, 2021.

PAPERS UNDER REVIEW

9. <u>Canh Minh Do</u> and Kazuhiro Ogata, "Parallel Specification-based Testing for Concurrent Programs", *Journal of Systems and Software*, 2020.

10. <u>Canh Minh Do</u>, Yati Phyo, Adrián Riesco and Kazuhiro Ogata, "A Parallel Stratified Model Checking Technique/Tool for Leads-to Properties", *ISSSR*, 2021.

11. Thet Wai Mon, Dang Duy Bui, Duong Dinh Tran, <u>Canh Minh Do</u> and Kazuhiro Ogata, "Graphical Animations of the NS(L)PK Authentication Protocols", *Journal of Visual Language and Computing*, 2021.

PENDING PAPERS

APPLICATIONS

These applications were developed when I have been doing research at JAIST. They are all freely available for use in the public domain.

1. **Spec-based**

   - An environment for testing concurrent Java programs based on specifications by using JPF and Maude.

2. **Jpf-parallelization**

   - Fully automate testing with JPF parallelization for enhancing software verification.

3. **L+1 Layers Divide & Conquer Approach to Leads-To and Eventually Model Checking**

   - A tool was developed in Maude on top of Full Maude for L+1 Layers Divide & Conquer Approach to Leads-To and Eventually Model Checking.

4. **Cafeobj VSCode Extension**

   - Visual Studio Code Extension for CafeOBJ language, includes syntax highlighting and snippets.

LANGUAGES

Proficient in English, basic Japanese and native Vietnamese.

REFERENCES

FROM ACADEMIA

Prof. Kazuhiro Ogata
Professor
Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi, Ishikawa 923-1292 Japan

Phone: +81-761-51-1211
ogata@jaist.ac.jp

FROM INDUSTRY

Mr. Nguyen Van Nam

CTO at Langmaster

485 Hoang Quoc Viet, Cau Giay, Ha Noi, VIetnam

Phone: (+84) 902-171-042

namnv@langmaster.edu.vn