

THÔNG TIN VỀ LUẬN ÁN TIẾN SĨ

1. Họ và tên nghiên cứu sinh: **Đào Thị Hương**..... 2. Giới tính: Nữ.....
3. Ngày sinh: 02/02/1983 4. Nơi sinh: Hải Phòng.....
5. Quyết định công nhận nghiên cứu sinh số: 1118/QĐ-CTSV ngày 24/12/2012 của Hiệu trưởng Trường Đại học Công nghệ.
6. Các thay đổi trong quá trình đào tạo:
Quyết định số 1049/QĐ-ĐT ngày 13/12/2013 của Hiệu trưởng Trường Đại học Công nghệ về việc thay đổi cán bộ hướng dẫn nghiên cứu sinh.
Quyết định số 959/QĐ-ĐT ngày 27/11/2015 của Hiệu trưởng Trường Đại học Công nghệ về việc gia hạn học tập cho nghiên cứu sinh Đào Thị Hương.
Quyết định số 47/QĐ-ĐT ngày 28/01/2016 về việc thay đổi đề tài luận án tiến sĩ cho nghiên cứu sinh Đào Thị Hương. Tên đề tài sau khi thay đổi: *Một số phương pháp kiểm chứng các hệ thống hướng đối tượng*.
7. Tên đề tài luận án: **Một số phương pháp kiểm chứng tái cấu trúc phần mềm**.
8. Chuyên ngành: Kỹ thuật phần mềm 9. Mã số: 9480103.01
10. Cán bộ hướng dẫn khoa học: PGS.TS. Trương Ninh Thuận
11. Tóm tắt các **kết quả mới** của luận án:

Đề xuất phương pháp bảo toàn bất biến trong tái cấu trúc biểu đồ lớp của UML.
Đầu tiên, luận án hình thức hóa mô hình phần mềm cùng với ràng buộc bất biến bằng ký pháp toán học. Kế tiếp, luận án định nghĩa khuôn mẫu (*template*) mô tả các phép toán tái cấu trúc. Tiến trình tái cấu trúc trên biểu đồ lớp được thực hiện bằng cách áp dụng các phép toán liên quan đến quan hệ phân cấp trong kế thừa, cụ thể là *Folding*, *Abstraction*, *Composition*, *Factoring* và *Unfolding*. Cuối cùng, luận án *xây dựng các luật tái cấu trúc* đối với từng phép toán và chứng minh sự đúng đắn của các luật này. Sự khác biệt của phương pháp đề xuất đối với các nghiên cứu trước đây thể hiện ở việc *xem xét và tích hợp các đặc tả về ràng buộc bất biến của lớp vào trong tiến trình tái cấu trúc*.

Đề xuất phương pháp kiểm chứng sự bảo toàn hành vi trong tái cấu trúc hệ thống phần mềm, có sử dụng mẫu thiết kế Strategy, tại các giai đoạn thiết kế và cài đặt của quy trình phát triển phần mềm. Quy trình kiểm chứng tính nhất quán bao gồm ba bước: (1)

Thực thi tiến trình tái cấu trúc, (2) Tính toán tiền/hậu điều kiện của các hệ thống trước và sau khi tái cấu trúc, và (3) So sánh, kiểm tra sự tương đương về ngữ nghĩa của các biểu thức mô tả ràng buộc hành vi của hai hệ thống. Tại giai đoạn thiết kế, mô hình hệ thống được biểu diễn bằng biểu đồ lớp và biểu đồ tuần tự, các ràng buộc về hành vi (tiền/hậu điều kiện) được biểu diễn bằng ngôn ngữ ràng buộc đối tượng OCL. Chú ý rằng, đối tượng được quan tâm đến sự bảo toàn về hành vi là các kịch bản (scenarios) tham gia vào tiến trình tái cấu trúc. Đây là điểm khác biệt trong đóng góp của luận án với các nghiên cứu trước. Tại giai đoạn cài đặt, quy trình kiểm chứng được thực hiện một cách tương tự tuy nhiên chế tác được thực hiện tái cấu trúc là bản cài đặt chương trình phần mềm. Luận án cài đặt chương trình bằng ngôn ngữ lập trình Java và mô tả các ràng buộc hành vi (bằng ngôn ngữ đặc tả JML). Kết quả cuối cùng về sự bảo toàn hành vi của hệ thống sẽ là sự kết hợp các kết quả kiểm chứng tại các giai đoạn thiết kế và cài đặt chương trình phần mềm.

Xây dựng công cụ CVT hỗ trợ kiểm chứng sự bảo toàn hành vi trong tái cấu trúc mô hình phần mềm. Công cụ này nhận các tệp dữ liệu đầu vào là mô hình ban đầu, mô hình sau tái cấu trúc, cùng với các ràng buộc về hành vi của chúng; dữ liệu đầu ra là kết luận về khả năng nhất quán giữa các mô hình này.

12. Khả năng ứng dụng trong thực tiễn: Các kết quả đã đạt được của luận án có thể được sử dụng trong các giai đoạn thiết kế, cài đặt và bảo trì của vòng đời phát triển của phần mềm nhằm đạt được mục tiêu cải thiện chất lượng của các hệ thống phần mềm.

13. Những hướng nghiên cứu tiếp theo: Nghiên cứu chuyên sâu các vấn đề về bảo toàn bất biến và hành vi trong tái cấu trúc hệ thống phần mềm. Cụ thể, đối với bài toán bảo toàn bất biến luận án xem xét kết hợp các tiến trình tái cấu trúc để tạo thành các chiến lược tái cấu trúc có ý nghĩa hơn đối với người dùng. Đối với bài toán bảo toàn hành vi nghiên cứu các kịch bản có chứa đựng các yếu tố phức hợp như các phân đoạn kết hợp (rẽ nhánh, lựa chọn, và lặp). Ngoài ra, có thể mở rộng bài toán kiểm chứng tính nhất quán đối với các hệ thống có các yếu tố chuyên biệt miền (hệ thống thời gian thực, hệ thống nhúng, các hệ thống đòi hỏi độ an toàn cao, v.v.).

14. Các công trình đã công bố có liên quan đến luận án:

1*. **Thi Huong Dao**, Hong Anh Le, and Ninh Thuan Truong. *An approach to analyzing execution preservation in Java program refactoring*. International Conference on Context-Aware Systems and Applications, pp.101-110, Springer, 2016.

2*. Hong Anh Le, **Thi Huong Dao**, and Ninh Thuan Truong. *A Formal Approach to Checking Consistency in Software Refactoring*. Mobile Networks and Application, Vol.22.2, pp.356–366, Springer, 2017. (ISI indexed)

3*. **Thi Huong Dao**, Thanh Binh Trinh, and Ninh Thuan Truong. *A Tool Support for Checking Consistency in Model Refactoring*. In Proc. of the 9th International Conf. on Knowledge and Systems Engineering, pp. 108-113, Springer LNCS, 2017.

4*. **Thi Huong Dao**, Xuan Truong Nguyen, and Ninh Thuan Truong. *Preservation of class invariants in refactoring UML models*. (Accepted by the International Journal of Software Engineering and Knowledge Engineering – IJSEKE 2018) (ISI indexed)

Ngày tháng năm 2018
Xác nhận của cán bộ hướng dẫn
(Kí và ghi rõ họ tên)

Ngày tháng năm 2018
Nghiên cứu sinh
(Kí và ghi rõ họ tên)

Trương Ninh Thuận

Đào Thị Hương

INFORMATION ON DOCTORAL THESIS

1. Full name: **Dao Thi Huong** 2. Sex: Female

3. Date of birth: 02/02/1983 4. Place of birth: Hai Phong

5. Admission decision number: 1118/QĐ-CTSV dated 24/12/2012.

6. Changes in academic process:

Decision No. *1049/QĐ-ĐT dated 13/12/2013* of the Principal of the VNU University of Engineering and Technology (VNU-UET) about change of Supervisor for Dao Thi Huong PhD student.

Decision No. *959/QĐ-ĐT dated 27/11/2015* of the Principal of the VNU University of Engineering and Technology (VNU-UET) about extend the period of study for Dao Thi Huong PhD student.

Decision No. *47/QĐ-ĐT dated 28/01/2016* of the Principal of the VNU University of Engineering and Technology (VNU-UET) about change the dissertation title to “*Approaches to verifying object oriented systems*”.

7. Official dissertation title: ***Approaches to verifying in software refactoring.***

8. Major: Software Engineering

9. Code: 62.48.01.03

10. Supervisor: Assoc. Prof. Trương Ninh Thuận.

11. Summary of the **new findings** of the dissertation:

Proposing an approach for preservation of class invariants in refactoring UML models. To achieve this aim, we first formalize the elements of UML class diagram, together with their invariants by making use of mathematical notations. We then introduce five operations that involve the hierarchy relationship. Furthermore, we also propose the refactoring rules in refactoring process and proving that these rules satisfy the conditions of preservation of class invariants. The difference of our proposed method from other studies is shown in the work of taking into account of class invariants in refactoring UML models.

Proposing an approach for verifying behaviors in refactoring software system at design and implementation phases of the software development life cycle. At the design phase, software model is represented by class and sequence diagrams, behavioral constraints including invariants, pre and post-conditions are expressed by Object Constraint Language (OCL). Based on the proposed consistent rules, the behavioral expressions of the evolution model are computed and used to compare with the behaviors of the original model. Note that we are particularly interested in the behavior preserving of the scenarios after refactoring process, this is a contribution of our research compared to the other research. At the implementation phase, the process of checking consistency is similar to the design phase excluding the difference of software artifact. The result of the behavioral preservation of the system is the combination of verification results at the design and implementation stages of the software system.

Developing the CVT tool support for checking consistency in model refactoring. This tool gets input data file as the initial model and its evolution along with their behavioral constraints and responses the conclusion about the potential consistency between these models.

12. Practical applicability, if any: The achieved results of the dissertation can be applied in the design, implementation and maintenance phases of the software development life cycle in order to reach the aim of improving the software systems' quality.

13. Further research directions, if any: Extensive study about the issues of invariants and behaviors preserving in refactoring software systems. Specifically, for the issue of preservation of invariants, we will consider the combination of five refactoring operations for the sake of making more meaningful jobs for the users. For behavioral preservation problem, we will study scenarios that contains complex aspects as combine segments (*alternatives, options, and loops*). Furthermore, the consistency verification approaches are possibility to extend for domain-specific systems (*e.g., real-time software, embedded software, safety-critical software, etc.*).

14. Thesis-related publications:

1*. **Thi Huong Dao**, Hong Anh Le, and Ninh Thuan Truong. *An approach to analyzing execution preservation in Java program refactoring*. International Conference on Context-Aware Systems and Applications, pp.101-110, Springer, 2016.

2*. Hong Anh Le, **Thi Huong Dao**, and Ninh Thuan Truong. *A Formal Approach to Checking Consistency in Software Refactoring*. Mobile Networks and Application, Vol.22.2, pp.356–366, Springer, 2017. (ISI indexed)

3*. **Thi Huong Dao**, Thanh Binh Trinh, and Ninh Thuan Truong. *A Tool Support for Checking Consistency in Model Refactoring*. In Proc. of the 9th International Conf. on Knowledge and Systems Engineering, pp. 108-113, Springer LNCS, 2017.

4*. **Thi Huong Dao**, Xuan Truong Nguyen, and Ninh Thuan Truong. *Preservation of class invariants in refactoring UML models*. (Accepted by the International Journal of Software Engineering and Knowledge Engineering - IJSEKE 2018) (ISI indexed)

Date: / /2018
Signature of supervisors

Date: / /2018
Signature of PhD student

Truong Ninh Thuan

Dao Thi Huong