Title	形式仕様と設計の整合性検証に関する研究
Author(s)	Vu, Dieu Huong
Citation	
Issue Date	2015-03
Type	Thesis or Dissertation
Text version	ETD
URL	http://hdl.handle.net/10119/12751
Rights	
Description	Supervisor:青木　利晃，情報科学研究科，博士

氏名	VU DIEU HUONG			
学位の種類	博士（情報科学）			
学位記番号	博情第 318 号			
学位授与年月日	平成 27 年 3 月 20 日			
論文題目	Study on Verifying the Conformance of a Design to Its Formal Specification （形式仕様と設計の整合性検証に関する研究）			
論文審査委員	主査	青木　利晃	北陸先端科学技術大学院大学	准教授
		平石　邦彦	同	教授
		緒方　和博	同	教授
		鈴木　正人	同	准教授
		石川　冬樹	国立情報学研究所	准教授

論文の内容の要旨

This work focused on development of reactive systems. A software development process begins with informal requirements which the target software is expected to meet. The informal requirements are translated into formal specifications to ensure their consistency. Then, system designs are developed as models for implementation. Finally, the implementation is done according to the designs using programming languages. In this development process, we should verify the fact that the designs satisfy the requirements described by formal specifications since incorrect designs likely lead to significant costs caused by back track of the developments. The specification captures the external behaviors including the results of the operations of the systems. Separately, the design represents the details of how to make the results. We consider that the formal specification languages such as Z, VDM, Event-B are appropriate to describe the specification because they provide rich notions (e.g., set, relation, and function) to facilitate describing the specification. They also provide tools to assure the consistency of the specification. Promela is an appropriate language for describing the design. In Promela, the design could be described in an imperative manner. Design decisions are straightforwardly described based on complex data structures (e.g, record type and array) and various control structures (e.g, selection and loop). Therefore, we intend to use Event-B and Promela for the specification and the design to facilitate describing them. Then, we propose a framework to verify the Promela design against the specification in Event-B. This framework is to verify the reactive systems. The first problem we must deal with is that there exists a gap between the specification and the design. The specification defines what behaviors are produced, whereas the design defines the detail of how the behaviors are produced. Since there exists such a gap, we intentionally use different specification languages: Event-B for the specification and Promela for the design. This in turn leads to the second problem; that is, we have to deal with difference of specification languages used for the specification and the design. Actions in Event-B are performed in parallel, whereas actions in Promela may

be performed step by step. Therefore, a state transition in the specification may be followed by multiple state transitions in the design. Another problem is that the reactive systems just operate if they receive stimulus from the outside, so-called environments. Therefore, the design must be verified in communication with the environment. The other problem is to assure the practicality of the framework. It must provide an ability to check important properties and detect typical bugs of the reactive systems. It is also possible for users to produce inputs of the framework. These must be demonstrated by some case studies including real systems.

The first contribution of the research is a new combination between Event-B and Promela/Spin included in a framework for the verification of reactive systems. This framework is to verify the conformance of the design to its formal specification where the design and the specification are described in different specification languages. Applying the framework, we can choose appropriate specification languages to describe the specification and the design for the purpose of verifying the design. With this combination between Event-B and Promela/Spin, we can check the design against the consistent and the correct specification. This would drastically improve the reliability of model checking results because the specification is reliable. The second contribution of the research is to fill the gap between the specification and the design. The specification defines abstract data structures, whereas the design defines implementable data structure. Also, the specification defines results of operations, while the design defines details of how to make the results. In the framework, we relate the specification to the design by common semantics, LTSs, and correspondences between state transitions given by mappings from syntactic elements in the former to those in the latter. This makes it possible to systematically verify the conformance of the design to the specification. The third contribution refers to supports for applying the framework to verify real systems. As mentioned, the users must produce the formal specifications in Event-B and the proper bounds for the verification of the system design. We give guidelines for translation from the informal specifications into the formal specification in Event-B. These facilitate the validation of the formalism. We also give a procedure to give the proper bounds to direct the verification focus on the behaviors relevant to intended properties and bugs. Therefore, we could determine appropriate bounds to avoid the state explosion when applying model checking; the critical cases could not be missed because we use proper bounds for the verification.

To evaluate the applicability and the effectiveness of our framework, we conducted some case studies in which the target systems are the reactive systems ranging from the simple systems to complex systems. Specifically, we applied our framework to verify a real system, an operating system compliant with the OSEK/VDX standard. The results of the several experiments are shown to demonstrate that this approach can be practically applied in verification of important properties and detection of typical bugs of the target systems. This exhibits an ability to deal with the specifications and the designs which are described in different specification languages. Therefore, we can choose appropriate specification languages to describe the specification and the design for the purpose of verifying the design.

Keywords: formal specification, design, formal verification, simulation relation, OSEK/VDX OS.

論文審査の結果の要旨

この博士論文では，車載 OS のようなリアクティブシステムを対象として，設計が仕様を満たすことを検証する一手法を提案している．提案手法では，仕様を形式仕様記述言語 Event-B により記述し，設計をモデル検査ツール Spin のための振る舞い記述言語 Promela で記述する．そして，モデル検査ツール Spin により，設計が仕様を満たすことを自動的に検証する．

この手法では，仕様と設計が異なる記述言語で記述されている．Event-B では，refinement と呼ばれる考え方があり，仕様から段階的に実装に近い記述を導出する方法がある．しかしながら，車載 OS のようなシステムでは，設計には，複雑なデータ型を用いて最適化された振る舞いが記述されているため，refinement により，仕様から設計を導出することは困難であった．一方で，Promela では設計を記述することは容易であるが，モデル検査ツール Spin で検証する際，時相論理により仕様に相当する性質を記述することが困難である．そこで，提案手法では，仕様と設計を，それぞれ，異なる記述言語である Event-B と Promela で記述する方式を採用している．

異なる記述言語 Event-B と Promela で記述された仕様と設計を検証する手法は提案されておらず，この博士論文では，それを可能とする形式的な枠組みを提案している．また，提案した枠組みに基づいて，Event-B で記述された仕様，Promela で記述された設計，それらを対応づける関係，および，探索境界を与えることにより，検証可能な Promela 記述を生成し，Spin により自動的に設計の検証が行えるツールを実装している．提案手法は，3 つの単純な例に適用した後，実際の車載 OS の設計に適用し，検証に成功している．車載 OS は，現在の車載システムを支える重要なソフトウェアであり，その重要性はますます高くっている．そのようなソフトウェアの信頼性を向上させる技術の提案であり，現実的で，有用性が高いと言える．また，異なる記述言語で記述し，検証を行う方式は，効果的であり，既存研究とは一線を画し，十分な新規性と独創性を持っている．

以上，本論文は，車載 OS のようなリアクティブシステムを対象として，その信頼性を向上させる手法の提案を行っており，学術的に貢献するところが大きい．よって，博士（情報科学）の学位論文として十分に価値があるものと認めた．