論文の内容の要旨

論文題目

Supporting Planning and Refactoring of Refinement Structure of Event-B Models
（Event-B モデルの詳細化構造の計画とリファクタリングの支援手法）

氏　名　小林 努

Systematic construction of highly reliable software systems is crucial.
Constructing formal specification is one of the most rigorous ways to that end, since it enables developers to thoroughly verify target systems from early phases of development.
Event-B, which is one of such methods, has been attracting much interest from academia and industry, because it supports a flexible refinement mechanism that mitigates complexity of constructing and verifying models of complex target systems by considering multiple abstraction layers of models.
Moreover, the refinement mechanism of Event-B enables developers to flexibly select what elements of the target system are introduced in each step of refinement (refinement structure).

Although the refinement mechanism supports comprehensible yet rigorous verification, there are difficulties in exploiting it in software development.
Firstly, although most studies on Event-B currently focus on reducing complexity of constructing models, models are used after its construction as well.
Constructed models should be maintained, and there is a strong demand to reuse a part of existing models to construct other models.
In the area of program code maintenance, highly automated methods and tools to support refactoring have been leveraged.
Methods similar to them should be provided for models of Event-B.
Secondly, in order to construct models, developers need to plan structure of refinement before constructing models.
Although this activity is advanced due to lack of rigorous representation of the specification, it is worth investigating how we can support it.
The space of reasonable plans is too large to grasp, and models constructed by following an intuition tend to cause inconsistencies and ineffective use of refinement mechanism.

Therefore, helping analysis of design spaces of reasonable refinement plans is important.
However, these problems are currently solved partially or only in specific domains, and there are no generic and systematic approach.

To address the problems above, we focus on the refinement structure and propose methods to explicitly handle refinement structure from the point of view of engineering.

As a preliminary work, we propose a generic and systematic view of refinement of Event-B models.
The problems above are generalized on the basis of the generic view and we provide a generic approach to handle the problem.

On the basis of this view, we provide methods to improve maintainability and reusability of existing Event-B models as an instantiation of the generic approach.
The method reconstructs the refinement structure of existing models by constructing models about different sets of variables than those of original models such that they keep consistencies defined in original models.
The core of the method is decomposing a refinement step by finding certain properties of models from existing models and finding additional properties from proof for existing models to make new models consistent with original ones.
By combining decomposing of refinements with composing of refinements, we provide a method to restructure a refinement chain according to given sets of variables to be considered in each step.

We also tackle the problem of planning refinement structure for a target system before constructing its models.
We view this problem as another instance of the generic problem, and propose methods to effectively search reasonable refinement plans and show comprehensible views of the solution space.
We define rationales of refinement structure to avoid invalid refinement and follow common refinement strategies in practice.
On the basis of the rationales, we propose a search method that effectively removes invalid and ineffective refinement plans and show solutions in a

comprehensible manner.

In our case studies to evaluate the proposed methods, we succeeded in decomposing large refinement steps in existing models, restructuring existing models to extract reusable parts for construction of other models, and planning reasonable and effective refinement plans from informal but structured description of target systems.

Considering the results and discussion on ways to elicit information that is necessary for our methods, we conclude that our methods can help developers to utilize Event-B and its refinement mechanism in software development.