

メタモルフィック・テストと要求工学

糸賀 裕弥[†]

要求仕様の抜けや誤りを発見する手法として、要求に対応したテストケースにもとづき要求仕様をレビューする方法がある。メタモルフィック・テスト手法をシナリオに適用し、要求定義におけるテストケース生成手法について検討した。

Metamorphic Testing and Requirements Engineering

Hiroya ITOGA[†]

In order to find out errors in software requirements specifications, it is useful to review the specification with some test cases on software requirements definition phase. In this paper, I consider a method to apply Metamorphic Testing to use cases to generate test cases for requirements review.

1. テストケースによる要求仕様の確認

要求定義はソフトウェア開発における最初の段階である。要求定義段階の成果物である要求仕様書をもとに、その後のすべてのソフトウェア開発が進む。要求仕様の品質は、開発されるソフトウェア製品の品質に大きな影響を与えるため、可能な限り要求仕様の抜けや誤りを発見・修正することが望ましい[1]。

これらの抜けや誤りに対する早期の発見・修正手法として、要求定義段階においてテストケースを作成し、要求仕様をテストケースにもとづきレビューする手法が用いられる[2]。要求仕様を精緻化していく際に、受け入れテストケースやシステムテストケースを作成し、システムの動き(テストケース)・その結果(期待される結果)を、ステイクホルダ間で確認することにより、要求仕様の精度を高める方法である。特に、利用者を想定したテストケースを用いて、要求の確認を行うことにより、具体的な要求の抜けや誤りの発見が期待できる。

本学で実施している産学連携による組込みソフトウェア開発プロジェクト演習においても、企業における経験にもとづき、要求仕様書の作成とあわせて、要求に対応したテストケースを記述することで、要求仕様書の抜けや誤りを発見させる演習を実施している。

しかしながら、要求定義段階での、あるいは要求仕様にもとづくテストケースの作成には困難が伴う。

当然のことだが、設計や実装は行われていないため、細かい仕様にもとづくテストケースの作成手、ホワイトボ

ックステストに類するテストケースの作成手法には期待できない。

また、受け入れテストやシステムテストの目的は、検証(verification)ではなく妥当性確認(validation)である。テストケースの作成には要求仕様のみ知識だけでなく、ソフトウェア製品の利用者の知識が必要となる。しかし、要求仕様の妥当性確認に利用可能な多くのテストケースとその結果を、ソフトウェア製品の利用者が作成することには期待出来ない。

すなわち、要求定義段階において、要求仕様あるいはそれに類する文書と利用者から得られた少ない知識をもとに、受け入れテストケースあるいはシステムテストケースと、期待される結果を生成する手法が求められている。本研究ではこのテストケースの生成手法において、メタモルフィック・テスト手法に着目する。

2. メタモルフィック・テスト

メタモルフィック・テスト[3]は、テストケースに対する期待される結果の判定問題の解決策である。すなわち、作成されたテストケースにもとづくソフトウェアのふるまいが、妥当であるのかそうでないのかを自動判定する手法である。

メタモルフィック・テストにおいては、テストケースにもとづくふるまいや出力が妥当であるかどうかを個別には判断しない。かわりに、あるテストケースで期待される結果と、テストケース間の関係にもとづいて同じ結果が期待される別のテストケースで期待される結果が同じ場合、それらの期待される結果の正誤にかかわらずテストが成功したとする。例えば三角関数 \sin の値

[†]立命館大学 情報理工学部 情報システム学科
Department of Computer Science, College of Information
Science and Engineering, Ritsumeikan University

を計算するソフトウェアのテストにおいて、 $\sin(29.8^\circ)$ の出力の値そのものの妥当性の正誤ではなく、得られた値と $\sin(29.8^\circ+360^\circ)$ が同じ出力を返せばテストが成功したと判断する。

同じ結果が期待されるテストケース間の関係をメタモルフィック関係と呼び、メタモルフィック関係にあるテストケースを増やすことで、テスト結果がより正解に近づくという性質を持つ。メタモルフィック関係は、上記のように数学的な性質などから得られる。

3. メタモルフィック・テスト・テストシナリオの生成

本研究においては、以下の手順で要求仕様を確認し、抜けや誤りを発見する。

- 1) 要求定義段階において記述されたシナリオ(ユースケース)を準備する。
- 2) シナリオどうしがメタモルフィック関係を満足するようシナリオを書き換える(テストケースの生成)。
- 3) 書き換えられたシナリオを用いて要求仕様をレビューする。

シナリオにおけるメタモルフィック関係は複数発見できると考えているが、現在はシナリオの事前条件・事後条件を用いる。すなわち、シナリオの事前条件・事後条件が変化しない書き換えにおいては、書き換え前後においてテスト結果が変化しないとする。これに対して、事後条件が変化する書き換えにおいては、テスト結果が必ず変化すると考える。

シナリオのイベント文を書き換えることによって、事前条件・事後条件がどのように変化するかを一般的に知ることは出来ない。本研究では格文法を用いた制限言語を用いたシナリオ言語と、イベント文による状態変化を解析することでこれを解決する[4]。

図 1 はテストケースの生成例を示す。斜体で表された部分は、窓際の席を指定できなかった場合のふるまいを示したイベント文列である。他のシナリオにも同様の書き換えを施すことで、窓際が満席である場合の結果どうし、窓際が空席である場合の結果どうしが、振る舞いが成功して同じであるかどうか、あるいは事前条件が異なる場合にふるまいが失敗するかどうかを確認することで、ふるまいの抜けや誤りを発見することが出来る。

4. 議論と今後の課題

本稿では、メタモルフィック・テスト・テスト手法を要求工学、特にシナリオによる要求獲得に適用し、要求仕様の抜けや誤りを防ぐ方法について検討した。

```
[新幹線の乗車券を購入する][お客]


```
[pre:窓際は 満席である]
{
 お客は システムに 指定券の購入を 伝える
 お客は システムから 乗車駅と下車駅を 聞かれる
 お客は システムに 乗車駅(品川)を 伝える
 お客は システムに 下車駅(新大阪)を 伝える
 お客は システムから 乗車可能な列車を 示される
 お客は システムに 希望の乗車時刻を 伝える
 お客は システムから 座席指定の方法を 聞かれる
 お客は システムに 希望の座席指定(窓際)を 伝える
 if(座席指定が 満席である ならば) then
 お客は システムから 満席を 示される
 お客は システムから 座席指定の方法を 聞かれる
 お客は システムに 希望の座席指定(通路側)を 伝える
 fi
 お客は システムから 座席番号を 示される
 お客は システムに 了解と 伝える
 お客は システムから 料金を 示される
 お客は システムに 料金を 支払う
 お客は システムから 切符を 受け取る
}
[post:] ← 窓際の切符を持っている状態にはならないはず
```


```

図 1:テストケースの生成例

シナリオにおけるメタモルフィック関係は事前・事後条件によるものだけではなく、特定の関係のあるイベント文どうしの生起順序の入れ換えや、主語や目的語にあたる名詞の入れ換えなどが考えられる。

今後は、ドメインごとに特有のふるまいの書き換えルールなどを準備しておくことで、より要求仕様のレビューに有用なテストケースの生成方法について検討するつもりである。

参考文献

- [1] 大西淳, 郷健太郎, 要求工学, ソフトウェアテクノロジーシリーズ, Vol. 9, 共立出版, 2002.
- [2] IPA SEC,高信頼化ソフトウェアのための開発手法ガイドブック, 2010. pp. 157-158. (東京海上日動システムズ株式会社の事例)
- [3] T. Y. Chen, S. C. Cheung, and S. M. Yiu, *Metamorphic testing: a new approach for generating next test cases*. Technical Report HKUST-CS98-01, Department of Computer Science, Hong Kong University of Science and Technology, Hong Kong, 1998.
- [4] 糸賀裕弥, 大西淳, シナリオにおけるアクタの状態の定義・参照関係を用いたふるまいの検証, FOSE2007, pp. 185-190.