

# ブラックボックス型のテスト設計技法の適用と自動化

張 曉 晶<sup>†1</sup>

テスト設計においてテスト技法を適用する難しさ、およびテスト技法を自動化する際の方針と2つの事例について議論する。

## How to Adopt and Automate Blackbox Testing Techniques

XIAOJING ZHANG <sup>†1</sup>

The authors would like to discuss on the obstacles to adopt the blackbox testing techniques within test design process, and the practices of automating such testing techniques by tools.

### 1. テスト技法とその適用障壁

テストはソフトウェア品質確保における重要な一手段である。開発現場では「テスト実行」の自動化に盛んに取り組んでいる一方、ソフトウェア仕様にに基づいたテスト項目の洗い出しを行う「テスト設計」は依然として属人的スキルに基づいて手作業で行うことが多く、稼働がかかるうえ、テスト項目の良し悪しが品質や納期にも大きく影響する。

効率的にテスト設計を行う方法としては、経験的に有用性が知られているテスト技法を利用することが考えられる。ISTQB ではテスト技法を大きく3つに分類している：(1) ソフトウェアの仕様にに基づくブラックボックス型のテスト技法、(2) ソフトウェアの内部構造に基づくホワイトボックス型のテスト技法、(3) バグが潜みそうな箇所を探る経験ベース型のテスト技法。筆者らはソフトウェア仕様の網羅を重視し、ブラックボックス型のテスト技法に着目する。ブラックボックス型のテスト技法には、同値分割/境界値分析/ドメイン分析テスト、AllPair 法、状態遷移テスト、デジジョンテーブルテスト、ユースケーステストなどがある。また、テスト技法には、様々な網羅基準 (coverage criteria) も合わせて提唱されている。状態遷移テストにおける状態網羅や遷移網羅がよく知られている例だが、Amman らの教科書<sup>1)</sup> では 34 種類もの網羅基準を紹介している。

テスト技法や網羅基準はその有用性が知られているが、適用するには一定の障壁を越えなければならない

という問題がある。経験的に、一般的なスキルレベルの開発者がテスト設計を行う際、多数存在するテスト技法や網羅基準を選択し手作業で適用すると、稼働がかかるうえ、過不足なくテスト項目を網羅的に洗い出すことが困難である。これは、以下の問題が存在するためである。

- 問題 1：適用可否の指針が無く経験と勘に依存  
どんな場面で、どの技法や網羅基準を選択し組み合わせればよいか判らない。
- 問題 2：曖昧性の無い、具体的な作業手順が不在  
技法を概念的に理解できても、あるいは、教科書の簡単な例題を解けても、実開発案件の複雑な例だと戸惑ってしまう。
- 問題 3：技法によっては煩雑な作業や計算が必要  
手作業では抜け漏れやミスが発生してしまう。

### 2. 自動化の方針と事例

テスト技法の適用障壁を低減し、テスト設計を効率化するため、筆者らは、Web 系業務システムの機能性テストをスコープとし、ソフトウェア設計書からテスト項目を網羅的に自動生成する取り組みを行ってきた。1 節で述べた適用障壁の問題点に対しては、それぞれ以下の方針で対応した。

- 対処 1：適用先を想定し明示する
- 対処 2：明確な定義づけを行い、アルゴリズムを定式化
- 対処 3：自動化ツールによる手作業の代替

対処 2 に関しては、テスト技法の本来のコンセプトや意図を理解した上で、曖昧性を排除して定式化する過程が特に肝要であると考えられる。論文や教科書などの

<sup>†1</sup> NTT ソフトウェアイノベーションセンター  
NTT Software Innovation Center

文献調査だけでなく、開発現場とも密に議論して、「準正常系」ケースを洗い出して綿密にアルゴリズムを構築する必要がある。対処 3 に関しては、テスト技法を自動化したツールを開発現場に導入するにあたり、既存開発プロセスとの親和性を意識する必要性に触れておく。多くの開発現場では、既に「開発標準プロセス」が規定されている。例えば、設計工程の成果物として、外部設計書、内部設計書、プログラム設計書など各種の設計書を作成し、またテスト工程の成果物として、テスト項目表などを作成することが定められている。ツール入出力としてこれらの成果物の形式や内容に準拠できれば、利用者は余分な学習や作業をする必要がないため、スムーズな技術導入に繋がる。以降、テスト技法を自動化した 2 つの事例を取り上げて、具体的に紹介する。

### 2.1 事例 1：ユースケーステスト技法

ユースケーステストは、ユーザシナリオやシステムの使い方に基づいてテスト項目を抽出するブラックボックス型のテスト技法であり、ユースケース記述で定められたフロー（ユーザとシステム間のインタラクション）を効果的に確認できる。対処 1 として、筆者らは本技法の適用先として、結合テスト工程の終盤ないしシステムテスト工程で行われるシナリオテストを想定し、明示した。シナリオテストはシステム全体をブラックボックスとしてとらえ、ユーザが特定の業務を遂行する流れを想定して確認を行うテストであり、Web システムであれば通常複数回の画面遷移を伴う。

ユースケーステスト技法は従来、ユースケース記述にある基本フローや代替フローを「意識して」テスト設計をするべきとのみ定められているため、具体的な作業手順が不在であった。そこで対処 2 として、筆者らは構造網羅や組合せ網羅といった網羅基準を取り入れて、より具体的に観点 A~D を定義し、これに沿ってテスト項目を網羅的に抽出する自動化手法を提案した<sup>2)</sup>。観点 A は基本フローを順次実行する。観点 B はそれぞれの代替/例外フローを 1 回のみ実行する。観点 C はなるべく全ての代替/例外フローを実行する。観点 D はそれぞれの代替/例外フローを複数回繰り返して実行する。アルゴリズム考案で「準正常系」ケースを考慮した一例を挙げると、ユースケーステスト技法の自動化において、代替/例外フローが計 1 つしか無いようなユースケース記述を考えると、観点 B と観点 C で重複したテスト項目が抽出される。そこで重複を許容して抽出したうえでテスト項目に重複ありという旨の印を付与し、出力可否を制御できるようにするという方式に決定した。

### 2.2 事例 2：ドメイン分析テスト技法

ドメイン分析テストは、同値分割/境界値分析を多変数（多次元）に適用したブラックボックス型のテスト技法であり、入力空間を効果的に確認できる。対処 1 として、筆者らは本技法の適用先として、結合テスト工程の前半で行われるパラメータテストを想定し、明示した。パラメータテストはシステムの 1 つの UI をブラックボックスとしてとらえ、ユーザが特定の入力を与えた場合の振る舞いを確認を行うテストであり、Web システムであれば Web 画面における入力バリデーションや次画面遷移までを含む。

ドメイン分析テスト技法は従来、依存関係をもつ複数の変数の定義域（ドメインと呼ぶ）の形状を意識して、同値分割によりドメイン内外の IN ポイント/OUT ポイント、境界値分析によりドメイン境界付近の ON ポイント/OFF ポイントを導出するとのみ定められており、テストに好適な性質を持つポイントを選択するための指針を欠いている。またそもそもとして、2 変数の場合はドメインを平面上に作図して各ポイントを求めるという煩雑な作業が必要であり、3 変数以上ともなるとドメインの形状をイメージしづらく手作業での導出は極めて困難となる。そこで対処 2 として、筆者らは、問題発生時の切り分けがしやすいという性質を考慮し、ポイントごとに「ポイントが満たすべき条件」だけでなく「互いに重ならない条件」も加えて制約式として立式し、それらを制約ソルバで求解してテストに用いる具体値を得る自動化手法を提案した<sup>3)</sup>。

## 3. まとめ

本ワークショップでは、テスト技法を適用する難しさ、自動化する際の方針と事例を紹介した。今後はテスト技法の適用がテスト設計の全体における位置づけをどのように明確に示すかについて、さらに検討を深めたい。

## 参考文献

- 1) Paul Ammann and Jeff Offutt. *Introduction to software testing*. Cambridge University Press, 2008.
- 2) 張曉晶, 丹野治門. 設計書に基づく web アプリのシナリオテスト向けテスト項目抽出手法. 電子情報通信学会技術研究報告. SS, ソフトウェアサイエンス, Vol. 115, No. 153, pp. 39-44, 7 2015.
- 3) 丹野治門, 張曉晶. ドメインテスト技法に基づく網羅的なテストデータ自動生成手法の提案. 情報処理学会研究報告. ソフトウェア工学研究会報告, Vol. 2014, No.6, pp. 1-8, nov 2014.