

SCORM 2004 教材における構造テスト支援ツールの検討

田中 頼人

早稲田大学 教育学部

eラーニングの学習の進捗に合わせて動作する適応型教材の効果や形式的記述方法に関する研究は従来からなされており、国際標準規格”SCORM 2004”として実用化が進んでいる。しかし、適応型教材の作成過程でテストを行い、作者の意図の通りに教材が動作するか否かを確認するための技術的な支援は、未だ十分ではない。制御構造を有する適応型教材は一般的なソフトウェアと同様に欠陥(バグ)を含む可能性を持ち、欠陥のある教材は本来の学習活動と関わりのないところで学習者に負担を強いてしまう。本発表の目的は、ソフトウェア工学の一分野であるソフトウェアテストの手法をeラーニング教材開発の手順の中で適用し、欠陥の少ない適応型教材の開発を支援するための技術的枠組について検討することである。

1. はじめに

eラーニング基盤技術の一つとして、ADLが策定したSCORM 2004[1]がある。この規格は学習者への個別適応機能を持ち、学習の進み具合に合わせた教材の動的な振る舞いを実現してきた。しかし適応型教材は制御構造を有するため、単なるテキストや画像、動画等の集まりではなくソフトウェアの一種と考えられる。教材作成には意図の通りに動作するかを確認するテストの必要性が生じるが、SCORM 2004教材に対するテストの環境は未だ十分とは言えない。本発表ではSCORM 2004の教材記述方法の特徴に基づいてテスト支援の現状を明らかにし、テストの一工程を対象とする支援ツールの提案を行う。

2. SCORM 2004 の特徴

SCORM 2004では、学習の順序立てや学習者の状況に応じた動的な振る舞いをif-thenルールの組合せによって教材中に記述することが可能である。また、「次の内容へ進む」「前に戻る」等のコマンドをLMSだけでなく教材側から発行できるようになり、学習者を導くための設計においても自由度が高い。これらの機能は「シーケンシング&ナビゲーション」と呼ばれ、SCORM 2004を強く特徴づけるものになっている。

SCORM 2004では教材作成者が教材の構造とその動作ルール(シーケンシングルール)をマニフェストファイルと呼ばれるXMLファイル中に記述することで、教材の振る舞いを制御する。LMSはシーケンシングルールを解釈し、提示画面を決定して学習者への配信を行う。

```
<item identifier="ITEM1" isvisible="true"
  identifierref="RES">
  <title>解説ページ</title>
  <imsss:sequencing>
  <imsss:sequencingRules>
  <imsss:preConditionRule>
  <imsss:ruleConditions conditionCombination="all">
  <imsss:ruleCondition
    referencedObjective="obj_exp1" operator="noOp"
    condition="satisfied" />
  </imsss:ruleConditions>
  <imsss:ruleAction action="skip" />
  </imsss:preConditionRule>
  </imsss:sequencingRules>
  <imsss:objectives>
  <imsss:primaryObjective satisfiedByMeasure="false"
    objectiveID="obj_exp1">
  <imsss:mapInfo targetObjectiveID="obj_x"
    readSatisfiedStatus="true"
    readNormalizedMeasure="true" writeSatisfiedStatu
    s="false" writeNormalizedMeasure="false" />
  </imsss:primaryObjective>
  </imsss:objectives>
  </imsss:sequencing>
  </item>
```

図1: 適応機能の例

図1はシーケンシングの典型的な使われ方である「プリコンディショナルルールによるSkip」のマニフェストファイルの例の一部を示している。学習者が解答していた事前テストの結果をreferencedObjective(学習目標)の値によって参照し、その結果が不正解ならLMSは解説ページを、正解なら解説ページをSkipし、次のページを提示する。これらの動作は教材作成者がマニフェストファイル中にシーケンシングルール、学習目標を組み合わせることによって、予め記述する。

上記のように、SCORM 2004の適応機能は、

- XMLファイルによる記述である
- If-thenルールの組合せである

- ・ 学習者との対話によって XML ファイルが解釈され、LMS は教材中の次の提示箇所を決定する

という特徴を持つため、教材中に欠陥を混入させないために教材作成者が行いうる施策の手順は次のような項目でモデル化できる。

1. マニフェストファイルが整形形式 (well-formed) であることの検証
2. マニフェストファイルが妥当 (valid) であることの検証
3. 学習の進捗に応じて変化するそれぞれの SCORM データモデルの値を、教材を実際に動作させて確認
4. 制御フローに着目し、マニフェストファイル中の命令や分岐が全て実行されるかどうかを確認
5. データフローに着目し、マニフェストファイル中で未定義のまま使用されているデータや未使用のままのデータ定義の箇所を特定
6. マニフェストファイルの論理構造は参照せず、入力と出力の関係に着目して動作を確認

上記項目は 1~2 が開発工程、3 がデバッグ工程、4~6 をテスト工程として分類可能である。教材をソフトウェアと見なす場合、4~5 は XML ファイルの内部構造に沿って行うため、ソフトウェア工学の用語を用いて「構造テスト」と呼べる[2]。同様に入力と出力の関係を検証する 6 は「機能テスト」である。

3. 支援ツールの現状

第 2 章で述べた手順 1~6 の中で「1. 整形形式であることの検証」は、W3Schools.com 等が提供するツールを用いて行うことができる[3]。「2. 妥当であることの検証」は Rustici Software 社が配布する文書型定義 (DTD) ファイル、および XML スキーマ定義ファイル[4]を用い [xmlvalidation.com](http://www.xmlvalidation.com) 等が提供するツールで可能である[5]。デバッグ工程である「3. SCORM データモデルの値の確認」は Trident IDE[6]のような市販の SCORM 2004 オーサリングツールで取得し、目視できる。しかし、テスト工程に相当する「4. 制御フローのテスト」「5. データフローのテスト」「6. 入出力関係のテスト」に関しては、現状では支援のためのツールが存在しない。

テスト工程へのサポートが十分に行われなければ、ソフトウェアとしての性質を持つ SCORM 2004 の教材は作成段階での誤りを発見することが困難となり、教材の信頼性を高めることができない。バグのある教材は本来の学習活動と関わりのないところで学習者に負担を与えるため、適応機

能による教材の高度化が進むようになると、教材に対するテスト工程は一層重要なものとなる。

4. 提案システム

本発表では、これまでに不十分だったテスト工程への支援の中で構造テスト、特に制御フローを対象としたツールの提案を行う。実行の手順は

1. マニフェストファイルの読み込み
2. 網羅したい要素を「命令」「分岐経路」「条件」から選択
3. 選択した要素を全て通る、論理構造上の経路を抽出
4. 抽出した経路を通る入力値を与え、テストを実行
5. 結果の確認

とする。実行には、マニフェストファイルを用いた教材実行が可能な SCORM 2004 シーケンシングエンジン[7]を用いる。同エンジンは筆者らが SCORM 2004 規格の普及を目的として開発を行ってきたもので、商用 LMS に組み込まれた実績や CD での教材配布に対応した実装例[8]がある。

5. おわりに

本発表では、ソフトウェアとしての特徴を持つ SCORM 2004 教材の性質をふまえて欠陥の除去に対する方策の現状を明らかにし、構造テスト支援ツールの設計方針について提案を行った。今後は支援ツールの実装を進めるとともに、SCORM 2004 教材の開発経験を持つ教材作成者コミュニティと連携し、支援ツールの試験運用による評価を行いたい。

参考文献

- [1] Advanced Distributed Learning Initiative, Sharable Content Object Reference Model (SCORM) 2004 4th Edition(2009)
- [2] 廣田豊彦, 橋本正明, プログラムの構造テスト支援, 情報処理学会ソフトウェア工学研究会報告, pp. 41-47(1998)
- [3] http://www.w3schools.com/xml/xml_validator.asp
- [4] Rustici Software, SCORM XML Schema Definition Files, <http://scorm.com/scorm-explained/technical-scorm/content-packaging/xml-schema-definition-files/>
- [5] <http://www.xmlvalidation.com/>
- [6] <http://www.scormsoft.com/trident>
- [7] 田中頼人, 樋田稔, LMS との連携が可能な SCORM 2004 シーケンシングエンジンの開発, 情報処理学会 CMS 研究発表会研究報告, pp. 51-55(2008)
- [8] 田中頼人, SCORM CD Maker: ネットワーク環境に依存しない SCORM 2004 教材配布システム, 教育システム情報学会研究報告集, Vol. 23, pp. 27-31(2009)