

日本語言語技術教育を取り入れた形式手法教授法

小林 健一[†]

株式会社豆蔵[†]

1. はじめに

形式手法を技術者が開発現場にて使用するうえで一番の困難は、「自然言語の表現を形式言語に翻訳する」作業である。

現在の形式手法研究では、自然言語にて表現された記述を形式言語に翻訳するのは「記述者の責任」としていることがある[1]。大学の教育カリキュラムでも、数理論理、各種形式手法の紹介が中心となっている[2]。自然言語の側面からの教育は不十分である[3]。

本稿では、形式手法の開発現場への適用方法を効果的に学習するためには、教材にどのような要素が必要かを明らかにすることを目的とする。

2. 実験

実験として、形式手法ツール Alloy Analyzer[4]の学習用教材を開発し、講座を実施した。

2.1 講座概要

- 受講対象者
形式手法に興味を持つソフトウェア技術者
- 前提知識
何らかのプログラミング言語経験
- 到達目標
 1. Alloy Analyzer を用いて、簡単なモデルを作成できる
 2. モデルに対する、簡単なプロパティを記述し、検証できる
- 講座スケジュール
 - チュートリアル 2 時間
 - 個人演習 2 時間

2.2 教材

教材は、講義テキスト、および演習問題からなる。

講義テキストは、Alloy Analyzer の特徴の解説、UML から Alloy Analyzer 仕様記述への変換の実演を含む。

演習問題の方針として、初期のデータ構造のみ、形式記述と可能な限り一対一対応するよう、日本語の問題文を構成した。この日本語の問題文を元に、形式手法ツールでの記述を学習する。この演習課題を通して、「日本語の曖昧さとは何か」を考えさせるようにした。

演習問題は、複数の Step に分かれる。学習者は Step 順に問題文を読み、Alloy Analyzer にて仕様を記述し、結果を確認する。

- Step1 :: データ構造以外の制約を持たない、単純な仕様の記述
- Step2 :: 不変制約の記述
- Step3 :: 反例を発見するための表明の記述
- Step4 :: 関数の定義
- Step5 :: 関数の事前、事後条件を検証する命題の記述
- Step6 :: オプション課題

2.3 演習の実施

演習教材は、個人演習用に開発した。ただし、演習においては隣同士の 2 人で 1 組になり、定期的に相談するように促した。これは、一人で長時間悩むことを避けるためである。特に実行結果の確認は、講師、および隣の学習者と共に行った。

演習中、講師は常に学習者の周囲を周回するようにし、こまめにツールの使用法を伝えた。初心者は、正しい結果が出ていても自信が持てないことが非常に多いためである。

3. 実験結果

- 受講者数 19 名
- 受講者職業
 - 技術者(企業) 53%
 - 研究者(企業) 21%
 - 教育・研究者機関 5%
 - その他

4. 結果考察

以下に、実施アンケートの抜粋、および演習実施を観察した様子を述べ、考察する。

4.1 講座全般について

表 1 期待した情報を得られましたか?

選択肢	選択数
そう思う	8
まあそう思う	10
どちらでもない	1
あまり思わない	0
思わない	0

表 2 講座の難易度はいかがですか?

選択肢	選択数
難しい	0
まあ難しい	12
適度	7
まあ簡単	0
簡単	0

4.2 受講者のアンケートコメント

- 「Alloy の本を読んだだけでは、モデリングの方法が良く分からなかったので、大変勉強になりました」
- 「これまで『抽象によるソフトウェア設計』を読んで、自分でできる気がしなかったのですが少しはできそうな気がしました。でも論理式で表現することや制約をみつけることに習熟しないと、実際に使いこなせないと思いました。効果的なテストを書ける能力と一致する部分が多いと思います(当然ですが、、、)」
- 「初めての形式手法でしたが、興味深く受講しました。ただ正直、消化不良なので『この次』をどうすればいいかなやましいです。(alloy 本も手をつけてますが、読みやすいとは必ずしも言えないです。。。)」
- 「演習に必要な知識が事前に input されない講義は辛い。日本語の Alloy もひどい出来だし、UML と Alloy の図の対応とかクラス図の Alloy への変換規則とか足りない説明沢山」

アンケートコメントを読んでいると、受講者の中に、文章をきちんと記述できていない者がいることに気づく。

不満を述べる文章であっても、きちんと記述できる者とできない者がいる。例えば上記コメントの「足りない説明沢山」として文章が終わる例である。

今回の講座では、演習結果とアンケートコメントの関係を記録していないが、不満を感じている者の多くは、演習を進められなかった要因として「日本語の文章力不足」があるものと思われる。

5. 関連研究

本稿同様に形式手法教育カリキュラムにて Alloy Analyzer を採用した研究として、中島氏の研究 [2]がある。

ソフトウェア関連文書の品質に関する研究として、小川氏らの研究 [5]がある。技術文書の評価を試みている。

形式仕様記述の言語を自然言語と結びつける研究としては、日野氏らによる、実行可能な形式化自然語 (FNL: Formalized Natural Language) の研究 [1]がある。自然語として英語を対象としている。

6. おわりに

今後の研究課題として、技術文書作成のための日本語能力とは何か、が挙げられる。現在、著者は、「言語技術教育」に注目している。特に、議論に焦点を合わせた教育を行うことが、形式手法学習に効果的ではないかと考えている。制約の検証を行うということは、反論をすることと同じだからである。

参考文献

- [1] 日野克重, 本位田 真一: “ソフトウェアの問題記述のための実行可能な形式化自然後の提案”, 情報処理学会論文誌 Vol. 52 No. 3 1365-1394 (Mar. 2011)
- [2] 中島 震: “形式手法教育カリキュラムでの数理論理の位置づけ”
- [3] “情報処理学会第 70 回全国大会シンポジウム情報専門学科におけるカリキュラム標準「J07」最終報告パネルディスカッション”, 2008
- [4] D. Jackson : Software Abstractions - Logic, Languages, and Analysis, The MIT Press 2006, [邦訳] 中島震(監訳), 今井健男, 酒井政裕, 遠藤侑介, 片岡欣夫(共訳): 抽象によるソフトウェア設計 - Alloy ではじめる形式手法, オーム社 2011.
- [5] 小川清, 斉藤直季, “ソフトウェア関連文書の品質改善”