

モデル駆動開発を利用したUMLモデリング学習支援

塚本 涼† 酒井 三四郎‡

静岡大学大学院総合科学技術研究科† 静岡大学情報学部‡

1. はじめに

近年のソフトウェア開発現場において、UML(Unified Modeling Language) はシステムの分析、設計の手段として広く利用されるようになり、モデリング教育の必要性が高まっている。しかし、指導者や時間のリソースが限られた大学の授業等を通じて初学者が十分なモデリング能力を身に着けることは困難である。その理由として、初学者にとって自身が作成したモデル図の正しさの検証が難しい事が挙げられる。

本研究では、学習者が自身のモデル図の妥当性を検証・改善できるようになることを目的に、モデル駆動開発(MDD: Model Driven Development)を用いて、モデルの動作状況を可視化させる学習支援機能を実現する。MDDによりモデルを動作させ、その実行ログを利用してモデルを検証し誤りの気づきを促進する。学習者の自己改善によりモデルの質が向上すれば、早期段階から指導者のより高次な指導が期待できるようになる。

2. 関連研究

MDDとは、作成したモデル(基本はクラス図とステートマシン図)からコードを自動生成する機能により、モデルを主体とした設計を可能とする開発手法である。コードの記述が不要なため、モデルの作成に集中でき、検証や再利用が容易になるメリットがある。MDD用のツールとして、BridgePointや、IBM社のRational Rhapsody等のツールが提供されている。MDDを用いたモデリング学習活動の研究として、モデルからロボット制御を行うコードを生成し、実際にロボットの挙動を観察することによりフィードバックを得る教育が行われている[1]。ロボットを必要とせず、視覚的に動作するシミュレータ環境で同様の実習を行った研究もある[2]。またMDDで重要となるステートマシン図の検証をするためのシミュレーションを行える学習支援ツールの開発が行われている[3]。

これら関連研究と本研究との主な相違点は、第一に、モデルからフィードバックを得る手段として、動作させた際の実行状況のログを利用するというア

プローチを取る点である。第二に、ロボットやシミュレータ環境の有無に依存せず、指定したフォーマットのログ出力機能を実装する事で利用可能な点である。

3. 提案システム

本研究で提案するシステムは、ChangeVision社のモデリングソフトウェアastah*のプラグインとして実装した。インターフェースを図1に示す。概略を示すと、図中①がモデル作成・編集エリアである。②は操作ボタンエリアであり、ログファイルの読み込み、ステップ実行、リセットボタンを配置している。③は遷移状況表示エリア、④は制約誤り診断エリア、⑤は制約設定エリアである。

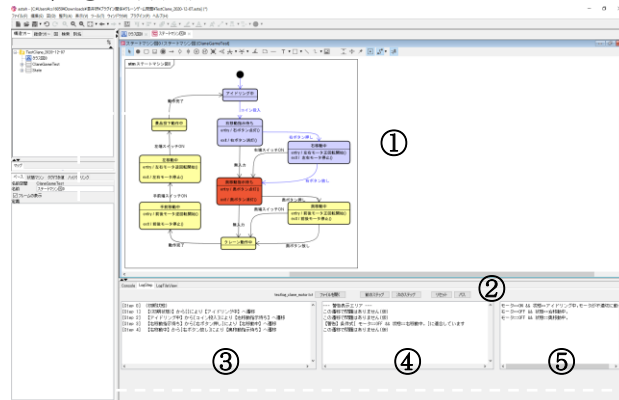


図1: システムのインターフェース

3.1 利用方法

本システムでは、MDDにより生成されたコードの実行時に、指定したフォーマットのログが出力され、それを読み込むことを前提とする。ログは、ステートマシン図における遷移毎に1行ずつ、状態とその状態に至ったイベント、およびその時点での各変数の値の情報が必要となる。すなわち、各状態における情報をログとして出力する仕組みが自動生成コードに組み込まれるようにする。コード生成・実行環境に制限はないが、本研究の開発環境としては、astah*で作成したモデルからコンソールで実行可能なコードを生成し、動作確認を行っている。この際、コード生成にはastah*プラグインm2t[4]を利用している。システム利用時は、ログファイル読み込みボタンから、出力されたログを読み込む。

3.2 ステップ実行機能

ステップ実行ボタンを押すことで、ステートマシン図における各状態に遷移した際の状況が確認でき

る。まず遷移状況表示エリア上では、どの状態からどのイベントによりどの状態へ遷移したか、という情報がテキストで表示される。モデル作成・編集エリアでは、各遷移に対しその状態と遷移が赤色でハイライトされる。以降、一度経由した状態と遷移は青色に変更される。未経由の状態と遷移は色が変わらないため、誤りが潜む可能性のある経路が発見しやすい。この様なステップ実行機能は、既存ツール[3]で実装されている機能と同様である。

3.3 制約誤り診断機能

変数と状態に関する条件式を設定し、条件式が満たされない場合に指定した警告文表示を行う機能である。一例として、変数“motor”が状態“Waiting”では“OFF”であると仕様から読み取れる場合に、制約設定エリアに「motor==OFF && 状態==Waiting: モータが不正動作」と入力する。すると、ステップ実行で状態 Waiting に遷移した際、もし motor が ON であれば、制約誤り診断エリアに“モータが不正動作”と表示され、学習者は仕様を満たさない不正な変数操作がどこかで行われている（もしくは必要な操作が行われていない）ことに気付く。条件式を満たしていれば警告は表示されない。

各状態における、各変数が取るべき（または取ってはならない）値を適切に条件式に設定しておくことで、学習者はモデルの作成段階やコードの実行段階では気付かなかった誤りの情報が明示的に与えられ、モデルの改善へ繋げることができる。また、警告文の内容を調整することで、学習者に提示する情報量を必要に応じて制御することができる。例えば婉曲的な警告表現に留めておくことで、学習者の思考へ負担を与え、モデルの理解を深めるきっかけとなることを見込める。

現状は指導者が各問題に対して事前に条件式を設定しておくことを想定しているが、学習者が仕様から変数が満たさなければならない条件を読み取る能力を会得するために、自身で考え設定させるという使用方法も考慮できる。

4. 評価と考察

本システムの評価を行うため、学習者の誤りを想定したケーススタディを行った。問題の題材は、UML モデリング技能認定試験のサンプル問題の中からクレーンゲームを取り上げた[5]。内容は、与えられた仕様書からステートマシン図を作成することとした。学習者によって、誤りを含む複数のステートマシン図が作成されると予想され、それら解答例に対してシステムを利用した際の有効性を検証する。

ステートマシン図作成における記法的誤りを除く誤り種類を整理し、本システムの利用による誤り発見容易度（不可、難、中、易の4段階）と、その発見方法について、ケーススタディによって検証した結果の概要を表1に示す。

表 1：ケーススタディ検証結果

ケース	誤り種類	発見容易度	発見方法
1	アクション位置誤り	易	制約誤り診断
2	アクション位置誤り	中	制約誤り診断
3	アクション記述不足	易	制約誤り診断
4	アクション記述不足	難	制約誤り診断
5	遷移不足	中	コード実行時
6	遷移先誤り	易	コード実行時

具体例を挙げると、ケース3では変数を操作するアクションをある状態に書き忘れた解答例で、その状態へのステップ実行時に警告が表示され誤りに気付くことができた。一方、ケース4も同様に記述不足の解答例だが、誤りのある状態へ至る複数の経路のうち特定の経路でしか警告（不正な変数値）が生じないため、実行時にその経路を辿っていなければ誤りに気付くことが困難であった。検証時、モデル作成・編集エリアでは未経由の状態と遷移が明らかでないため、それを元に全ての経路を実行・検証すれば発見できる誤りであるが、図の複雑さ次第では現実的に手動での検証は難しい場合があり、現状の機能では効率的に誤りの発見ができるとは言えない。ケース5では、遷移の不足があった場合、コード実行時に不足しているイベントを発生させれば動作が変化しないことから誤りに気付くが、イベントを能動的に発生させなければ気付くことができない。

ケーススタディにより、このツールによって発見できる誤りが判明し、モデルの改善に繋がる気付きを与える一定の効果が期待できることが確認できた。一方で、誤りの発見が困難な状況も判明した。

5. おわりに

本研究では、MDD を利用したモデリング学習支援システムの開発を行い、一定の有効性が見込めることを確認した。現時点では一部の誤りの発見が難しいという課題点やUI上の問題があり、改善の必要がある。今後更なる支援機能の考案、実装を行いたい。また、実装した機能を用いた効果的な学習支援教材を作成し、実際に学習者に利用してもらい評価実験を行いたい。

参考文献

- [1] 香山 瑞恵, 小形 真平, 永井 孝: モデル駆動開発方法論に基づく UML プログラミング教育環境, 教育システム情報学会誌, Vol. 36, No. 2, 2019, pp. 118-130.
- [2] 柴田 敦也, 武苗 棟之, 神原 弘: UML エディタからのコード生成によるロボット制御実習, 2017 年度 情報処理学会関西支部 支部大会 講演論文集, 2017.
- [3] 小形 真平, 岡野 浩三: Smart-Learning: DSL ベースの UML ステートマシン図学習支援ツール, 研究報告ソフトウェア工学 (SE), 2016, pp. 1-6.
- [4] astah: モデル駆動開発 m2t プラグイン, <https://astah.change-vision.com/ja/feature/astahm2t.md-plugin.html> (accessed 2020-12-15).
- [5] UML: UML モデリング技能認定試験サンプル問題, https://umtp-japan.org/about_exam/exam_sample (accessed 2020-12-15).