

オブジェクト指向モデリング教育のための オブジェクト図自動生成システムの設計と評価

早川 勝¹ 野沢 光太郎¹ 松澤 芳昭^{2,a)} 酒井 三四郎^{2,b)}

受付日 2012年3月19日, 採録日 2012年10月10日

概要: オブジェクト指向モデリング教育支援を目的として, クラス図に対応したオブジェクト図を自動生成し学習者に提示するシステムを開発した. 本システムが想定する2つのユースケースは, (1) 学習者が作成したオブジェクトモデルの検証, (2) クラスモデルの多重度誤りの発見, である. オブジェクト図の生成アルゴリズムについては, あるクラスのインスタンスを1つ生成し, そのインスタンスの関連先のクラスのインスタンスについて, 多重度制約を考慮して無作為抽出し生成する. 大学院生13名を対象として, 本システム利用有無による対照実験を行った. 被験者には, 継承を含む3関連, 5クラスで構成される程度のモデルに対して, 本システムが想定する2つのユースケースに関する問題が与えられた. 実験結果は, (A) 実験群の正答率が統制群を上回った, (B) 実験群の解答プロセスでは想定されたユースケースが機能していた, (C) 本システム利用の時間コストは無視できる程度に小さい, であり, 提案システムの有効性が示された.

キーワード: UML, モデリング, オブジェクト図, CASE, 学習支援

Design and Evaluation of Object Diagram Automatic Generator for Object-Oriented Modeling Education

SHO HAYAKAWA¹ KOTARO NOZAWA¹ YOSHIAKI MATSUZAWA^{2,a)}
SANSHIRO SAKAI^{2,b)}

Received: March 19, 2012, Accepted: October 10, 2012

Abstract: The authors have developed a system which can generate sample object diagrams from a class diagram in order to support learners of object-oriented modeling with UML. The system supports two use-cases for learners: (1) supporting verification of an object model created by learners, and (2) supporting identifying inconsistencies between a class model and a specification. The system generates object diagrams in the following algorithm. First the system creates an instance of the class, then the system randomly decides the number of instances associated with the generated instance within multiplicity constraints, and generates them. The authors conducted a control experiment in which thirteen graduate students divided into two groups where only the experimental group could use the proposed system to solve a given task. The tasks given to learners require relevant skills for the assumed two usecases, and the model used in the task is composed of five classes and three relationships including one inheritance. The results were summarized as (A) average score of the experimental group was better than that of the control group's, (B) by qualitative analysis of the learners' answering processes, we confirmed the two usecases functioned effectively, (C) the effort to use the system was minimal enough to not be detrimental to the process, hence the effectiveness of the system has been verified.

Keywords: UML, modeling, object diagram, CASE, learning support

¹ 静岡大学大学院情報学研究科
Graduate School of Informatics, Shizuoka University, Hamamatsu, Shizuoka 432-8011, Japan

² 静岡大学情報学部
Faculty of Informatics, Shizuoka University, Hamamatsu, Shizuoka 432-8011, Japan

^{a)} matsuzawa@inf.shizuoka.ac.jp

^{b)} sakai@inf.shizuoka.ac.jp

1. はじめに

近年のソフトウェア開発言語の高水準化によって「モデリング」教育の必要性が高まっている。企業情報システム開発においては、オブジェクト指向に基づくUML (Unified Modeling Language) を用いた設計が世界標準となっている。2000年以降、モデルベース開発 (MDA: Model Driven Architecture または MDD: Model Driven Development) および支援ツールの実用化が進んでおり、普及段階にある。これを受けて、モデリングの国際会議 MODELS では教育者会議が行われており、モデリング教育の必要性が議論されている [1]。わが国でも MDD に対応した先進的教育の試みが報告され始めている [2]。

本論文で議論の対象とするのは、構造モデリング (UML のクラス図、オブジェクト図によって表記されるモデル) である。構造モデルは UML のセマンティクスアーキテクチャにおいて最下層のレイヤに属し、他の振舞いや相互作用モデルの基礎となるモデルであることがその理由である。構造モデリングにおいては現実世界 (モデル化対象) をオブジェクト指向の側面から解釈し、クラス図として表現することが主要な作業項目である。

オブジェクト指向設計・モデリングの教育は難しいといわれている (2章で先行研究を示す)。その理由の1つとして、モデリングには1つとなる正解の定義が難しいことがあげられる。児玉は「概念の再構築がモデリングである」と述べている [3]。つまり、モデル化対象を「概念」として解釈し、「分類」「命名」を行って世界を再構築する必要がある。その世界が第三者に伝達可能かどうかを検証する必要がある。このような理由で、解説書などに載っている演習問題を解くというような一般的な学習方法をとることができないのである。

現在確実に有効なモデリング教育方法は、モデリング成果物を指導者が添削することであるが、大学の一斉指導方式では、1人1人の学習者の成果物に対して指導者が木目細かな添削を行うことは困難である。この問題を解決する方法として、グループ内のコミュニケーションの促進を支援すること [4] や、学生同士のピアレビュー方式 [5] などが提案されている。しかしながら、それらの方式は正確に評価されておらず、効果は未知数である。筆者らの経験では、初学者はクラス図に記述された第三者のモデルの制約 (入門レベルでは多重度) を正確に理解してコミュニケーションすることが困難である。基礎能力がない学習者のピアレビューは表層的なものになることが危惧される。

そこで、本研究ではオブジェクト図に着目したモデリング初学者のクラス図読解支援を考えた。オブジェクト図は、クラス図の表現しうる一場面を表現する図であり、クラス図と比較してモデルの意図が曖昧にならず、具体的な読解ができることに着目したからである。このことについ

て、Chiorean ら [6]、城之内 [7]、上原ら [8]、児玉 [3]、などがオブジェクト図を使った教育を提案している (これらは、2.1 節で紹介される)。

本研究のテーマは、クラス図に対応したオブジェクト図を自動で生成する (以降、本システムが生成したオブジェクト図を「オブジェクト図サンプル」と呼ぶ) ことでクラス図読解を支援することである。生成されたオブジェクト図サンプルを学習者に提示することで、学習者が生成したオブジェクトモデルを自ら検証したり、クラスモデルの多重度誤りに気付くことを促進したりする。自己評価によって学習者の初期成果物の質を高めることができれば、ピアレビュー方式の学習効果向上や、指導者によるより高次*1の添削実施が期待できる。

本論文は全7章からなる。2章で先行研究のレビューを行う。3章では、筆者らが提案するシステムの目的とユースケースについて述べた後、システムの基本設計、アルゴリズム、およびインタフェースについて述べる。4章では評価実験の方法、5章ではその結果を報告する。6章で考察を行い、7章で成果をまとめる。

2. 先行研究

本章では、1) モデリング教育手法、2) モデル評価支援、3) モデリング学習支援ツール、の各観点から先行研究をまとめて述べる。

2.1 モデリング教育手法に関する研究

本研究は、モデリング教育手法に関する以下の研究に立脚している。

Kuznaiarz らは、学生に対して UML およびオブジェクト指向ソフトウェア開発を教育するための授業の設計を行い、学生たちの学習効果を上げるためのポイントを8つあげている [9]。学生の学習効果を上げるうえで、特に効果があったポイントとして、(1) 授業の過程を通してすべての成果物の一貫性を保証させること、(2) 2回の反復作業により成果物の質を向上させること、の2つを取り上げている。本研究では、(1) について、クラス図、オブジェクト図の一貫性の保証を支援し、(2) について、学習者のクラス図読解による、反復作業の支援を行う。

オブジェクト図を利用したモデリング教育手法がいくつか紹介されている。城之内 [7] は、UML を用いたモデリングによる設計はトップダウン的になり抽象度が高く、初学者にとっては「とっつきにくい」とし、授業の中でクラス図を導く際にはオブジェクト図を利用している。支援ツールの項で言及する奥平ら、上原らの研究 [8]、[10] においても、オブジェクト図を記述することでクラス図の確認する方法が提案されている。児玉の入門書 [3] では、オブジェ

*1 例として、命名の善し悪し、要求仕様の充足度、結合度・凝集度の観点からの設計添削などがあげられる。

クト図を用いてクラス図の誤りを見つける方法が紹介されている。平沢の入門書 [11] では、オブジェクト図と同様にオブジェクトの構造を示す「ベン図」が書籍全般にわたって説明に利用されている。Chiorean らも、多重度や OCL (Object Constraint Language) で記述された制約に従った正確な読解をするためには、クラス図のスナップショット (つまりオブジェクト図) が有効ではないかという考察をしている [6]。このように、オブジェクト図を利用したモデリング教育手法そのものに新規性はないが、入門書でもオブジェクト図は取り上げられていないものも多くあり (たとえば文献 [12])、十分に一般的とはいえない状況と考えられる。オブジェクト図を利用した教育の有用性の測定、支援環境の提案については、2.3 節で比較する奥平らの研究 [10] が唯一である。

2.2 モデル評価支援に関する研究

オブジェクト図を明示的に使用していなくても、クラス図を扱うことで暗黙的にオブジェクトモデルを扱うことになる。クラス図教育のためのモデル評価に関する研究との関連を述べる。

Hasker らはクラス図およびユースケース図に含まれる欠陥の発見を自動で行うシステムを開発している [13], [14]。Hasker らは、「多重度の付加における誤り」や「複数のクラスとコンポジションの関連を持つクラスが存在する」、「継承において、継承元と継承先の関係が逆になっている」などの、クラス図によく見られる欠陥を定義した。UMLint [13] は、それらの欠陥がクラス図内に含まれているかどうかを解析し、欠陥が含まれている場合はユーザにその旨の提示を行うシステムである。このシステムは現在 UMLGRADER [14] として改良されている。UMLGRADER は、学生が作成したクラス図によく見られるクラス図の欠陥について、対応する欠陥のパターンを限定することで、より具体的な助言をすることを目的としている。類似する研究として、Auxepaules らは異なる 2 つのクラス図を比較して相違点を検出する手法を提案している [15]。相違点検出の目的は教育利用であり、熟練者のモデルと学習者のモデルの比較によってモデルの良さを評価しようという試みである。これらのアプローチは、「正解となるクラス図」が設定されている点に本研究との相違点がある。筆者らは、クラス図の正解は一意にならず、「正解」を求めるような教育はモデリング教育に適さないと考えている。この理由で、筆者らはモデル間の不整合の有無をモデルの評価基準として利用することを基本方針としている。

杉浦らは、伝統的なモデル評価の尺度である凝集度と結合度に基づいて、初学者がモデルの良さを評価できる指標を提案している [16]。その支援ツールである Focus は初学者がクラス図やコミュニケーション図に対して修正を加えるたびに凝集度および結合度を再計算し提示を行う。本研

究は、凝集度・結合度のようなデザインレベルのモデルの良さを議論するために、その前提となるモデル間の不整合がないモデルの記述を支援する。

2.3 モデリング支援ツールに関する研究

最後に、これまで提案されているモデリング支援ツールを取り上げ、本研究との関連を述べる。

ツールのユーザインタフェースの改良による学習支援研究がある。Auer らは、教育用 UML モデリング支援ツールとして、UMLet を開発している [17]。Auer らは、UMLet の特徴として、(1) Java アプリケーションとして開発することで、動作環境に依存しない、(2) 直感的に操作可能なインタフェース、(3) 成果物を一般的な形式で出力できる、の 3 点をあげている。Scott らは、教育用の設計支援ツールとして minimUML を開発している [18]。minimUML の特徴は、クラス図のみ作成することができ、扱う図要素もクラス、インタフェース、関連、集約、実現のみに限定している点であるとしている。提供する機能や扱う要素を限定することで、初学者にも扱いやすいツールにすることを目指したと Scott らは主張している。本研究は、上記の研究とは根本的に異なり、作図の直接支援機能と、図間の整合性に着目した支援を指向している。

Dranidis らは、教育用の設計支援ツール StudentUML を開発している [19], [20]。StudentUML の特徴は、図間の一貫性チェックを行う機能を提供していることにある。たとえば、シーケンス図で、クラス図にないクラス名を利用している場合に警告を行う。一貫性チェックのアプローチの問題点は、クラス図そのものの誤り (要求仕様との不整合がある) に対応できない点と、複数の異なるオブジェクトの状態の網羅性を担保できないところが問題点であると、筆者らは考えている。

金城らは、学習者に、第三者が作成したモデル図のインスペクションを行わせることで、モデル図の読解力を向上させることを狙い、学習者が実施したインスペクション結果と熟練者が実施したインスペクション結果の比較による学習法の提案とその支援ツール開発を行っている [21], [22]。クラス図の読解を指向しているという目的は類似しているが、このアプローチの問題点は、比較対象となる熟練者のインスペクションデータがないと機能しない点であり、限定された問題でのみしか適用できない。

奥平ら、上原らはオブジェクト図を自動生成するツールを提案している [8], [10]。上原らは、クラス図を理解する上で、オブジェクト図の利用の有効性を主張しており、2 つのクラス間の関係に着目し、その関係を示すオブジェクト図を生成する機能を提案している。本研究のアプローチのアイディアは、これらの研究と同様である。しかしながら、奥平らの研究の問題点として、1) オブジェクト図生成アルゴリズムが部分的にしか示されておらず、かつ、多重

度を考慮したオブジェクトの数の選択方式がユーザによる「半自動」方式である、2) 機能が提案の段階であり実装が成功した報告はなく、実験はシステムが生成したことを見立てた用紙を配布する方式で行われている、3) 正解となるクラス図のクラス数が2と非常に限定された問題でのみしか実験を実施していない、4) オブジェクト図の提示により学習者によるクラス図の修正活動が行われたことは確認されているが、正解から不正解への改悪もみられるなど最終的に正答率が向上していない、があげられる。本研究の新規性は、上記の問題点を改良し、実際にソフトウェアを開発し、継承を含む5クラスレベルのモデルで実証実験を実施している点にある。

本研究と奥平らの研究のもう1つの相違点は、奥平らの研究では、「クラス図を描く」、「描いたクラス図を確認する」ことを焦点として実験が行われ、「クラス図の読み」は暗黙的に扱われているのに対して、本研究では「クラス図の読み」に焦点を当てて実験を行っていることがある。この違いは、奥平らの研究は、学習者が記述したクラス図とオブジェクト図の変化に注目したプロセス分析の結果を示しているのに対して、本研究では、ツールが生成したオブジェクト図と学習者が作成したオブジェクト図の照合過程に着目した結果を示したという違いがある。本研究が着目した照合過程のモデルを次の3.1節に示す。

3. システムの設計

3.1 システムの目的とユースケース

本システムは、任意のクラス図を入力として、オブジェクト図サンプルを出力する。学習者が、出力されたオブジェクト図サンプルと、クラス図を読解し記述（または想像）したモデル*2を比較することで、クラス図読解の正しさを検証できるようにすることが本システムの目的である。

この目的を詳しく定義し、評価可能なように以下の2つのユースケースを提示する。ここで「ユーザ」は学習者を指し、「UC」はユースケース（UseCase）を表す。

UC1 学習者作成のオブジェクトモデル検証支援

UC2 クラスモデルの多重度誤りの発見支援

UC1（学習者作成のオブジェクトモデル検証支援）についての説明図を図1に示す。入力はクラスモデルであり、ユーザ（学習者）はこのクラスモデルを読解し、オブジェクトモデルを生成することが課題である。クラスモデルから出る破線矢印はオブジェクトモデルの作成を表す。オブジェクトモデルAはユーザ（学習者）が作成し、オブジェクトモデルBはシステムが作成する。このとき、学習者が

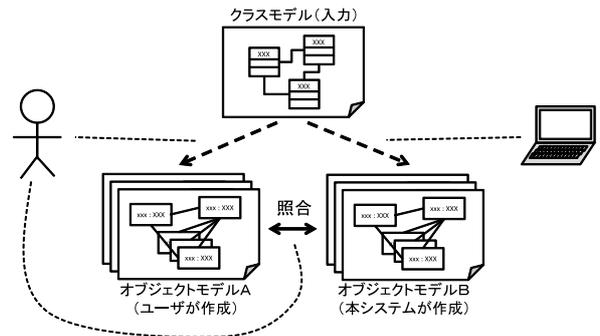


図1 UC1: 学習者作成のオブジェクトモデル検証支援

Fig. 1 The model of usecase1: supporting verification of an object model created by learners.

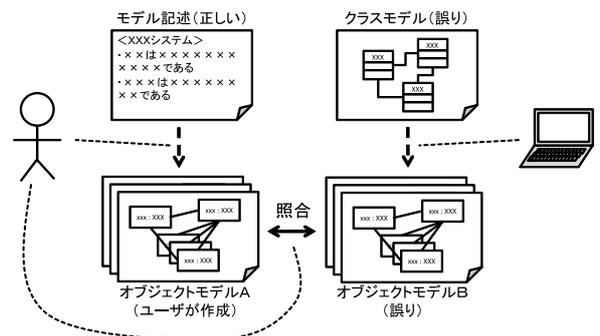


図2 UC2: クラスモデルの多重度誤りの発見支援

Fig. 2 The model of usecase2: supporting identifying inconsistencies between a class model and a specification.

作成するオブジェクトモデルAについて、実際に紙や計算機上に描かれておらず、学習者の想像上のみある場合も含む。

学習者はオブジェクトモデルAとオブジェクトモデルBを照合する。結果に不整合がある場合、クラスモデルとオブジェクトモデルBの一貫性は保証されていることから、誤りはオブジェクトモデルA（クラスモデルの読解）か、クラスモデルそのものにあることが分かる。照合の結果に整合性が認められた場合、クラスモデルが正しく読解できたことを確認できる。

UC2（クラスモデルの多重度誤りの発見支援）についての説明図を図2に示す。入力は誤りのあるクラスモデルであるが、クラスモデル単体では正誤は決まらないため、モデル記述（自然言語によってシステム仕様を説明したもの）も入力とし、それが正しいものとして、クラスモデルとの不整合を誤りとする。学習者はこの入力を読解し、誤りを訂正することが課題である。

学習者は、モデル記述から手動生成したオブジェクトモデルAと、本システムが自動作成したオブジェクトモデルBを照合し、不整合を確認することでクラスモデルの誤り発見をすることができる。クラスモデルとオブジェクトモデルBの一貫性は保証されていることから、オブジェクトモデル上の不整合の原因は、クラスモデルの誤りか、オブ

*2 ここでの「モデル」とは、クラス図やオブジェクト図における、MVC (Model-View-Controller) アーキテクチャでの Model を意図している。本論文では「クラスモデル」や「オブジェクトモデル」という用語を同様の意味で使用する。たとえば「クラス図」を View としたとき、その実体を表現する Model を「クラスモデル」と呼ぶ。

ジェクトモデル A の作成ミスに絞ることができる。双方を手動で作成した場合には、クラスモデルとオブジェクトモデル B の矛盾の可能性が否定できなくなるため、問題の特定が難しくなる。照合の結果に整合性が認められた場合、クラスモデルは誤りであることからオブジェクトモデル A の作成に誤りがあることを特定できる。

3.2 インタフェース

本システムは、ChangeVision 社のソフトウェア設計支援ツール、astah*のプラグインとして実装した。ChangeVision 社は astah* API を提供している。これを利用することで、astah*で編集されたクラス図に対して、astah*で編集できるオブジェクト図の自動生成が可能である。

本システムのユーザインタフェースを紹介する目的で、本システムのスクリーンショットを図 3 に示す。図内の、数字を記入した場所の名称と機能を以下に列挙する。

- (1) 拡張タブビュー。この拡張タブビューの中央にあるボタンがオブジェクト図サンプルの自動生成を実行するボタンである（以下、オブジェクト図サンプル生成ボタンとする）。
- (2) 図ビュー。クラス図や生成したオブジェクト図サンプルはここに表示される。
- (3) プロジェクトビュー。同一プロジェクト内に保存されている図やモデルがツリー状に表示される。

図ビューにてクラス図を選択している状態で、拡張タブビューにある「オブジェクト図サンプル生成ボタン」を押下すると、選択しているクラス図に対応したオブジェクト図サンプルを生成し、図ビューに追加表示する。その際、生成元のクラス図を左側、生成したオブジェクト図サンプルを右側に並べて表示する。これはクラス図とオブジェクト図サンプルを同時に参照できるようにするためである。

プロジェクトビューには、生成するたびに、そのオブジェクト図サンプルが追加されていく。過去に生成したオブジェクト図サンプルを再度見たい場合には、プロジェクトビューから選択し、図ビューに表示することができる。オブジェクト図をそのまま編集し、配置変更などを行うこ

ともできる。

3.3 オブジェクト図サンプル生成アルゴリズム

本節では、本システムのオブジェクト図サンプル生成アルゴリズムを説明する。オブジェクト図サンプル生成アルゴリズムは「基本アルゴリズム」「クラス図が3クラス以上からなる場合」「クラス図に継承が含まれる場合」の3つのアルゴリズムから構成されるため、その順に従って説明する。

3.3.1 基本アルゴリズム

以下に、基本となる「2クラス1関連」からなるクラス図から、オブジェクト図サンプルを生成するアルゴリズムを述べる。

- (1) クラス図にある関連を取得する。
- (2) 取得した関連から、一方の関連端のクラスを選択する。
- (3) 選択したクラスのインスタンスが0でなければ1つ生成し、1以上であれば生成するかどうかを無作為に決定し生成する。
- (4) (3) で生成した場合は、他方の関連端のクラスのインスタンスの数を確認し、0であれば生成するインスタンスの数を多重度の範囲で無作為に決定し、生成する。
- (5) (3) で生成したインスタンスと(4)で生成したインスタンスをリンクする。
- (6) (2) から(5)の手順を、(2)で選択しなかった方の関連端のクラスについて、再度行う。

図 4 に、本システムを利用して生成したオブジェクト図サンプルの例を示す。図 4 中、左上が対象とするクラス図である。本システムでは、クラス図が表現しうる様々な場面をオブジェクト図サンプルとして示すことができるが、出力しうるすべてのオブジェクト図サンプルを本論文で例示することは困難である。したがって、例として3つのオブジェクト図サンプルを示す。出力オブジェクト図サンプル A、B、Cはそのクラス図が表現しているモデルから出力されるオブジェクト図サンプルの例である。出力オブジェクト図サンプル A は、「1人の学生が1つの大学に属している」場面を描いたオブジェクト図、出力オブジェ

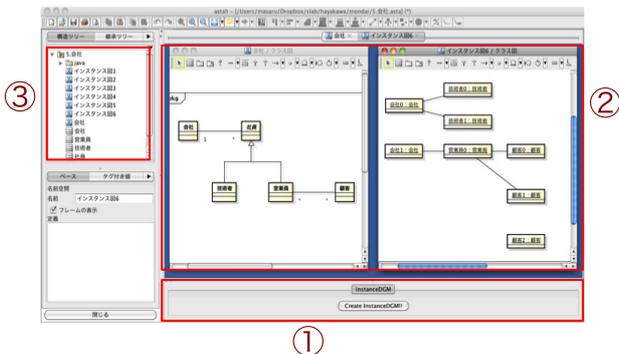


図 3 本システムのユーザインタフェース
Fig. 3 User interface.

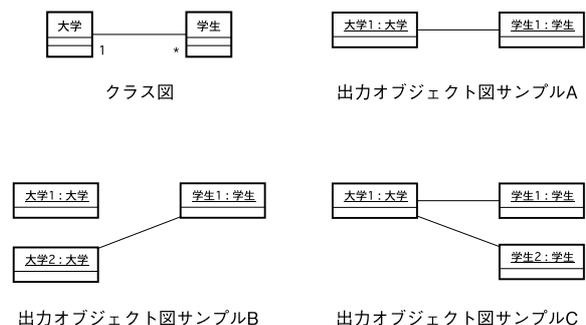


図 4 生成されるオブジェクト図サンプルの例 1
Fig. 4 Example of object diagram sample (1).

クト図サンプル B は、「1 人の学生が 1 つの大学に属している」と「学生が属していない大学がある」場面を描いたオブジェクト図、出力オブジェクト図サンプル C は、「2 人の学生が 1 つの大学に属している」場合のオブジェクト図である。

上記の手順 (4) にある「多重度に応じた」について補足説明をする。本アルゴリズムでは、多重度が「*」または「0..*」であれば、0 個、1 個、2 個のいずれかの個数のインスタンスとリンクする。多重度が「1..*」であれば 1 個もしくは 2 個のインスタンスとリンクする。どの個数になるかは、無作為に決定する。また、多重度が「1」のときには必ず相手方のインスタンスを生成し、リンクする。

3.3.2 3 クラス以上からなるクラス図に対応するアルゴリズム

クラス図に 3 クラス以上のクラスが存在する場合でも、基本的には 3.3.1 項に示したアルゴリズムを適用する。ただし、複数のクラスと関連を持つクラスがある場合、3.3.1 項の手順 (5) において、インスタンスがすでに存在することがある。この場合、既存のインスタンスとリンクをするか、新しくインスタンスを生成しリンクするかを無作為に決定する。存在しない場合は、多重度に依りて、無作為に決定した個数のインスタンスを生成し、リンクする。

図 5 に 3 クラスで構成されるクラス図からのオブジェクト図サンプル生成例を示す。図 5 中、左上の図が対象のクラス図である。出力オブジェクト図サンプル A, B, C はそのクラス図から生成されるオブジェクト図サンプルの例である。出力オブジェクト図サンプル A は、「授業を履修していない学生がいる」、「履修している学生がいない授業がある」および、「1 人の学生が 1 つの授業を履修している」場面を描いたオブジェクト図、出力オブジェクト図サンプル B は「履修している学生がいない授業がある」、「1 人の学生が 2 つの授業を履修している」場面を描いたオブジェクト図、出力オブジェクト図サンプル C は「2 人の学生が履修している授業が 1 つある」場面を描いたオブジェクト図である。

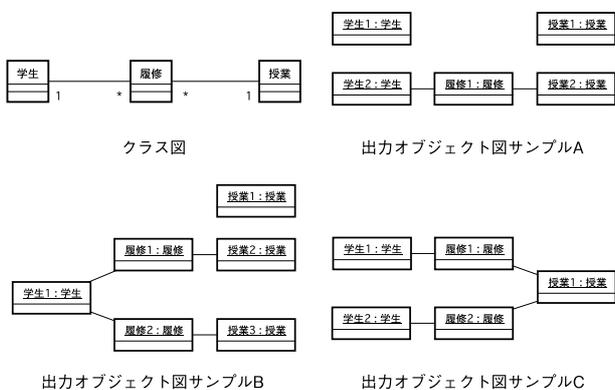


図 5 生成されるオブジェクト図サンプルの例 2
Fig. 5 Example of object diagram sample (2).

この例においては、関連を「学生-履修」、「履修-授業」に分割して考える。先に「学生-履修」側の関連について 3.3.1 項に示したアルゴリズムでオブジェクト図サンプルの生成を進めると、「履修-授業」の関連について同様の手順を進めたとき、「履修」のインスタンスがすでに存在していることになる。この場合、「履修」のインスタンスを新たに生成しリンクするか、既存のインスタンスに対してリンクをするかは無作為に決定する。

しかしながら、オブジェクト図生成の実行が後になった関連 (例の「履修-授業」) において、複数のクラスとリンクを持つクラスからインスタンスが新しく生成されると、先のオブジェクト図生成を実行した関連 (例の「学生-履修」) に付加されている多重度によっては、オブジェクト図にモデルとの矛盾が起こりうる。そうした矛盾が起こらないようにするために、クラス図とオブジェクト図間の矛盾を毎回検証し、矛盾が発生した場合は、インスタンスを追加して矛盾を解消する。

3.3.3 継承が含まれるクラス図に対応するアルゴリズム

クラス図内に継承が含まれる場合についても、3.3.1 項に示した手順 (3) までは同様である。以下では、手順 (3) で関連端のクラスにサブクラスが存在する場合、それ以後のアルゴリズムを説明する。

- (1) 関連端のクラスを継承しているサブクラスを取得する。複数であった場合は、そのうちの 1 つを無作為に決定する
- (2) (1) で取得したクラスが抽象クラスであるかどうかを確認する。抽象クラスの場合、取得したクラスのインスタンスは生成しない。
- (3) 以下、3.3.1 項に示した手順 (4) 以降と同様。

図 6 に継承が含まれる 3 クラスで構成されるクラス図からのオブジェクト図サンプル生成例を示す。左上の図が対象のクラス図である。出力オブジェクト図サンプル A, B, C はそのクラス図から生成されるオブジェクト図サンプルの例である。出力オブジェクト図サンプル A は、「営業員の社員が 2 人いる会社」と「営業員の社員が 1 人いる会社」

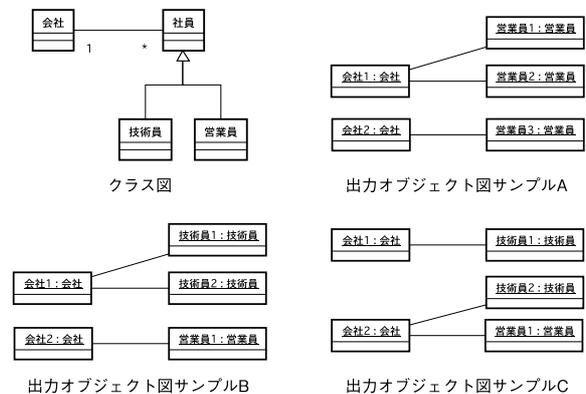


図 6 生成されるオブジェクト図サンプルの例 3
Fig. 6 Example of object diagram sample (3).

を描いたオブジェクト図，出力オブジェクト図サンプル B は「技術者の社員が 2 人いる会社」と「営業員の社員が 1 人いる会社」を描いたオブジェクト図，出力オブジェクト図サンプル C は「技術者の社員が 1 人いる会社」と「営業員と技術者が 1 人ずついる会社」を描いたオブジェクト図である。

A, B, C は，インスタンスの数やそのリンクのされ方が似ているが，「会社」のインスタンスが「営業員」のインスタンス 2 つとリンクされているケースや「技術者」のインスタンス 2 つとリンクされているケース，「技術者」と「営業員」のインスタンス 1 つずつとそれぞれリンクされているケースなど，クラス図が表現しているモデルを多面的に表現したオブジェクト図が生成されている。

4. 実験方法

本章では，本システムの評価実験の方法について述べる。

4.1 実験課題

本システムの評価のために実施する課題は，UML モデルリング技能認定試験入門レベルの問題集 [23] を参考に筆者らが作成したものを使用した。以下に，実験に使用した例を示す。以下に示すのは問題 1 である（実験に使用した全問題文を付録に示す）。

問題 以下のモデルを読み，次の 2 つの設問に解答しなさい。

- (1) クラス図に存在する誤り（モデル記述と矛盾する多重度）を見つけて修正しなさい。
- (2) クラス図修正後のモデルについて，典型的な場面を表現するオブジェクト図を 1 つ描きなさい。

モデル化対象 ある大学での授業履修システムである。

モデル記述

- 学生は複数の授業を履修できる。
- 履修を希望した授業は，必ず受講できるものとする。
- 学生は，同じ授業を再履修することも可能である。

クラス図



実験は 2 つの課題に分けられる。クラス図中に含まれる，モデル記述と矛盾する点を発見し，修正する課題（以下，クラス図修正課題と呼ぶ）である。もう 1 つは，与えられたモデルの典型的な場面を表現したオブジェクト図を描くという課題である（以下，オブジェクト図作成課題と呼ぶ）。以下，それぞれの課題について上記の例を用いて説明する。

4.1.1 クラス図修正課題

クラス図修正課題は，UC2（クラスモデルの多重度誤りの発見支援）の有効性の検証を目的とする。図 2 に示した

ように，正しいとするモデル記述と誤りのあるクラスモデルを入力とし，クラスモデルをモデル記述に矛盾しないように多重度もしくは関連の修正を行うことが課題である。

問題 1 を用いて，システムを利用した場合に想定される被験者の解答プロセスを述べる。被験者は，モデル記述から紙や計算機上または想像上にオブジェクトモデル A を作成する。その後，システムを利用しオブジェクトモデル B を生成する。オブジェクトモデル A とオブジェクトモデル B を照合し，1 つの授業を複数の生徒が履修することや，1 人の学生が同じ授業を再履修することができないことに気づく。その結果，被験者は修正箇所が「授業」から見た「履修」の多重度であることを特定し，「1」から「0..*」または「*」へ正しく修正を行う。

システムを利用しない場合，被験者はオブジェクトモデル B も自分で作成する必要がある。または，オブジェクトモデルを作成せずクラスモデルとモデル記述を直接比較する。このとき，正しくオブジェクトモデルが作成できなかったり，クラスモデルを正しく読解できなかったりする場合「同じ授業を再履修できない」ことを発見することができない。このように，修正箇所の特定が困難になることが想定される。

4.1.2 オブジェクト図作成課題

オブジェクト図作成課題は，UC1（学習者作成のオブジェクトモデル検証支援）の有効性の検証を目的とする。オブジェクト図作成課題は，クラスモデルを入力とし，被験者がそのクラスモデルの典型的な場面を表現したオブジェクト図を作成する課題である。

オブジェクト図作成課題の正答は一意に定まらないので，評価基準について詳しく述べる。本課題における「典型的な場面」とは，クラス図から想像できる一般的な例のことである。多重度が「*」や「1..*」のようにその範囲に幅があるもの場合は，複数例を用いて，モデルから表現されるパターンをできるだけ 1 つのオブジェクト図内で，網羅するように被験者に求めた。ただし，今回の実験ではパターンが網羅されていなくても，1 パターン以上が表現されており，かつオブジェクト図がクラス図と矛盾していなければ正答とした。クラス図とオブジェクト図の矛盾とは，インスタンスのリンク数が対応する多重度の範囲を逸脱，クラス図で登場していないクラスのインスタンスが登場していることである。

システムを利用した場合に想定される被験者の解答プロセスを述べる。被験者は，図 1 におけるクラスモデルからオブジェクトモデル A を紙上や想像上で作成する。次にシステムを利用し，オブジェクトモデル B を生成する。オブジェクトモデル A とオブジェクトモデル B を照合し，不整合がないか検証する。不整合があれば，もう 1 度オブジェクトモデル A かクラスモデルを見直す。不整合がないことを確認した場合は，そのオブジェクト図を解答とする。

オブジェクトモデル A が想像上で作成されていた場合、被験者はオブジェクトモデル B として作成したいくつかのオブジェクト図サンプルの中から、典型的な例に近いものを選び、解答のひな形として使用することも想定する。典型的な例に一致する出力があった場合にはそのまま解答として利用してもよい（ただし、インスタンス名は変更する必要がある）。いずれにしても、被験者が解答として利用できるかを検証するプロセスが含まれるため、生成されたオブジェクト図サンプルが（そうした検証なしに）解答として提出されることは想定しない。

システムを利用しない場合、被験者はオブジェクトモデルの検証をクラスモデルとの直接比較によって行う必要がある。このときどのような認知操作が行われるかについては不明であるが、システムを利用した場合のオブジェクトモデルどうして比較する方法よりも、高負荷であることが予想される。

4.2 実験計画

実験は難易度別の 3 問について、システムを利用して問題を解く実験群と、システムを利用しない統制群を用いた計画で実施された。

4.2.1 問題の難易度と出題順

実験に用いた問題の概要を表 1 に示す。問題は両群で、問題 1 から 3 まで昇順に解答する。問題 1 から順に難易度が上昇する。たとえば、問題 1 から 3 までクラス数、関連数が増加し、問題 3 では関連に継承が含まれる。問題の難易度による被験者の正答率の変化を比較することで、本システムの適用範囲を明確にすることが目的である。問題の順序による学習効果については両群同条件であるため比較した際の結果に影響しないと考えた*3。

4.2.2 被験者の選定と振り分け方法

本研究のユーザとして対象としているのは、大学の授業で UML を利用したモデリングの講義を受けたことがある初学者である。本実験は、大学院の授業でモデリングを受講中の 13 名を対象に行った。

被験者はランダム・サンプリング法を用いて実験群 7 名と統制群 6 名に振り分けられた。さらに、振り分けられた被験者が等質化されているかどうかを確認するために、事前試験を行った。事前試験の正答率は、実験群が 85.7%で

統制群が 100%であった。この結果より、被験者の等質化はなされていると判断した。

事前試験の課題について説明する。モデル記述、場面記述（自然言語によってシステムのユースケースのある一場面を記述したもの）を入力として、クラス図とオブジェクト図を作成するものである。この課題の目的は「実験群と統制群の等質を確認すること」、のほかに「実験において被験者が Astah を利用した作図作業がスムーズに実施できるようにすること」がある。本課題は、両群で本システムを使用しない。制限時間は無制限（1 週間以内の任意のタイミングで解答できる）とした。課題の難易度は、クラス数 3、関連数 2 程度で解答できるものである。評価基準は、(1) オブジェクト図が場面記述の内容を表現していること、(2) クラス図とオブジェクト図との間に矛盾がないこと、の 2 点である。事前試験に使用した問題文を付録に示す。

4.3 実験の手順

本実験の手順を以下に示す。括弧内の数字は要した時間である。

- (1) 実験の説明 (10 分)：デモ動画を用いて本実験の内容、課題の解法や実験中のスクリーンキャプチャソフトによる録画方法について説明を行う。
- (2) システムの使い方の説明 (5 分)：デモ動画を用いて、実験のすすめ方、および本システムの Astah への導入方法と利用方法について説明を行う。
- (3) 実験 (30 分)
- (4) アンケート (10 分)：本システムに対する使い勝手や印象に関するアンケート

実験手順に関する補足事項は以下のとおりである。

- 解答は、与えられた Astah ファイルを編集することで行う。与えられるファイルには、修正するクラス図があらかじめ記載されている。
- 各問の制限時間は 10 分とし、延長は行わない。
- 実験群と統制群の両方について、各問の終了直後に正解は提示しない（3 問終了後、全問の正解を提示する）。

4.4 実験の記録方法

本実験の記録方法を以下に示す。

- (1) 実験中のコンピュータ操作のスクリーンキャプチャソフトによる録画記録
- (2) 実験監督者による観察記録

5. 実験結果

5.1 正答率と解答時間

各問題における正答率と平均解答時間を表 2 に示す。列に実施した問題、行に群をとる。列にある「修正」はクラス図修正課題、「作成」はオブジェクト図作成課題を表す。クラス図修正課題が誤答であった場合、作成したオブジェ

表 1 出題問題の概要

Table 1 Questions for the experiment.

問題番号	出題内容
問題 1	3 クラス, 2 関連からなるモデル
問題 2	4 クラス, 3 関連からなるモデル
問題 3	5 クラス, 3 関連 (継承を含む) からなるモデル

*3 授業の一環として実施したため、学習効果も得られる設計としている。

表 2 正答率と解答時間

Table 2 Percentage of correct answers and average answer time.

		問題 1		問題 2		問題 3	
		修正	作成	修正	作成	修正	作成
実験群 (n=7)	正答率	57.1%	57.1%	71.4%	71.4%	85.7%	71.4%
	平均時間	9:02		7:48		8:38	
統制群 (n=6)	正答率	33.3%	33.3%	33.3%	33.3%	66.7%	50.0%
	平均時間	8:18		8:50		7:46	

クト図がクラス図と矛盾していても誤答とした。つまり、「作成」の正答率はクラス図修正課題，オブジェクト図作成課題の両方を正答した割合を示している。

問題 1 から 3 を通して「修正」，「作成」とともに実験群が高い正答率を示している。問題 3 におけるクラス図修正課題の正答率は，両群とも他の 2 問に比べ高い値を示している。クラス図修正課題を正答した者のみで集計したオブジェクト図作成課題の正答率は，問題 1，問題 2 では両群ともに 100% であり，問題 3 では実験群が 83.3% で統制群が 75% である。

平均解答時間について，統制群に対する実験群の比率を検証した。問題 1 では約 1.1 倍，問題 2 では約 0.9 倍，問題 3 では約 1.1 倍であった。筆者らによる観察結果においても，システム利用につまずいている被験者は見られなかった。これらをふまえて，本システム利用における時間的コストは，全体の学習時間と比較して無視してよい程度に小さいと考えられる。

5.2 システムの利用頻度

実験群における被験者ごとのシステムの利用状況を表 3 に示す。「時間」はクラス図修正課題，オブジェクト図作成課題の双方の解答を記入するまでの時間である。「達成」は○であれば正答したことを示し，×であれば誤答であったことを表す。「生成回数」は当該の課題中に被験者がシステムを利用してオブジェクト図サンプルを生成した回数を表す。このとき，被験者のオブジェクト図サンプル生成の目的が「修正」と「作成」の双方に該当することがあったため，その場合は「修正」，「作成」どちらの回数にも加えている。

実験全体での平均生成回数は 2.4 回であった。筆者らの想定では多重度のパターン数を網羅する 4 回程度を見込んでいたが，それよりも少ない値であった。この理由について，本節の以降の段落と，5.3 節で考察する。

平均生成回数は，0.8 回から 4.3 回と個人差がある。理由は 2 つ考えられる。(1) 本システムには無作為にオブジェクト図サンプルを生成するという性質があること，(2) 被験者個人にとっての問題の難易度，である。修正課題では，矛盾が発見されるようなパターンオブジェクト図サンプルが生成されるまでの生成回数が無作為になる。被験者がオブジェクト図サンプルを生成するまでもなく修正箇所

表 3 システムの利用状況

Table 3 Learners' usage status of the system.

被験者		問題 1		問題 2		問題 3		平均生成回数
		修正	作成	修正	作成	修正	作成	
A	時間	9:30		9:00		10:00		1.5
	達成	×	×	×	×	○	×	
	生成回数	1	2	2	1	2	1	
B	時間	9:40		7:00		9:30		2.3
	達成	×	×	○	○	○	○	
	生成回数	2	1	3	3	2	3	
C	時間	8:30		9:00		7:00		2.3
	達成	○	○	×	×	○	○	
	生成回数	4	1	4	0	3	2	
D	時間	9:30		7:40		9:30		2
	達成	○	○	○	○	○	○	
	生成回数	1	1	2	1	3	4	
E	時間	8:40		9:00		9:50		0.8
	達成	×	×	○	○	×	×	
	生成回数	0	0	1	3	1	0	
F	時間	10:00		6:10		6:30		4.3
	達成	○	○	○	○	○	○	
	生成回数	10	3	5	2	3	3	
G	時間	7:30		6:50		8:10		3.3
	達成	○	○	○	○	○	○	
	生成回数	4	3	2	4	2	5	
平均生成回数		3.1	1.6	2.7	2.0	2.3	2.6	2.4

気づいていた場合，生成回数は少なくなる。

問題ごとの平均生成回数は，クラス図修正課題では 2.3 回から 3.1 回，オブジェクト図作成課題では 1.6 回から 2.1 回と個人の平均生成回数に比べ，差は微小である。問題に登場するクラス数や関連数といったクラス図の要素が増えると，作成課題での平均生成回数が増えている傾向がある。理由としては，クラス図の要素が増えることでオブジェクトモデルのパターンも増えるため，照合をする回数が増えたこと，被験者が想像上でオブジェクトモデルの検証を行うことが難しくなったことが考えられる。クラス図の要素が増えると，被験者はクラスモデルと整合性のあるオブジェクトモデルを作成できる自信がなくなっていくと推察する。したがって，クラス図の要素が増加するにつれ，本システムは有効になると考えられる。

生成回数が 0 回のケースが 4 例，1 回のケースが 10 例ある。生成回数が 0 回の場合の理由として，(1) 被験者にとって問題が簡単でありシステムを利用する必要がないと判断した，(2) システムを有効に利用できる場面が分からず利用しないまま解答を終えてしまった，が考えられる。生成回数が 1 回の場合の理由として，(1) 修正課題において 1 回の生成で矛盾を発見できた，(2) 作成課題において 1 度生成して作成したオブジェクトモデルに自信を持った，(3) 試しに 1 度だけ生成をしたがシステムの有効性を理解できなかった，が考えられる。上記のいずれの理由においても，その要因の 1 つとして利用方法の説明不足が考えられるため，利用方法説明の改善によって生成回数の増加が見込まれる。

正答率と平均生成回数に相関が見られる。被験者 A と E の平均生成回数は，それぞれ 1.5 回と 0.8 回である。この

表 4 UC1 と UC2 における成功例と失敗例

Table 4 Succeeded case and failed case on UC1 and UC2.

	成功例	失敗例
UC1 (学習者作成のオブジェクトモデル検証支援)	11	1
UC2 (クラスモデルの多重度誤りの発見支援)	5	2

2名は、正答率が他の被験者と比べて低い。逆に被験者 F と G の平均生成回数は、それぞれ 4.3 回と 3.3 回であり、両者とも全問正答している。

問題の解答時間と生成回数には相関は見られなかった。このことは、5.1 節で述べた本システムの利用における時間的コストは無視してよい程度に小さいという結果を支持している。

5.3 個別の解答プロセスの分析結果

実験群の個別の解答プロセスについての質的分析を行い、UC1, UC2 それぞれに対して筆者らが想定したプロセスによる解答が行われているかを検証した。その結果、UC1, UC2 に対するそれぞれ成功例と失敗例の数の合計が得られた。それらをまとめたものを表 4 に示し、それぞれの実例について、以下に述べる。

5.3.1 作成課題に成功したケース (被験者 G, 問題 2)

作成課題に成功したケース (被験者 G, 問題 2) について述べる。被験者 G は、解答を作成する前に、システムを利用して 4 回ほどオブジェクト図サンプルを生成している。生成されたオブジェクト図サンプルを 1 つずつ吟味した後、生成されたものの中から最も典型的な場面を表現しやすいものを選び、それをひな形として、解答を作成している。解答の作成時、生成した他の 3 つのオブジェクト図サンプルを見ながら、典型的なものになるようにインスタンスを追加している。その後、インスタンス名をより具体的なものに変更している。実験後のアンケートによると被験者 G は、「生成したものの中で、典型的な場面を表現しやすいようなものを基にして、オブジェクト図を作成した」と回答している。このことから、被験者 G の頭の中に典型的な場面を表現したモデルが最初に存在していたと考えられる。このモデルは、図 1 でいうオブジェクトモデル A にあたる。被験者 G は、生成されたオブジェクト図サンプルを吟味していたことから、オブジェクトモデル A と生成されたオブジェクトモデル B を比較し、照合を行っていたと考えられる。

被験者 G のように、作図する前にオブジェクト図サンプルを生成するというケースは、他に 9 例存在した。この 9 例すべてにおいて、生成したオブジェクト図サンプルをそのまま解答とする者はおらず、被験者 G の例のように、生成したオブジェクト図サンプルを吟味していることが確認された。

5.3.2 修正課題に成功したケース (被験者 F, 問題 3)

修正課題に成功したケース (被験者 F, 問題 3) について述べる。問題 3 のクラス図修正課題の正答は、「顧客」と「社員」の関連を削除し、「顧客」と「営業員」に関連を引き、多重度はどちらから見た場合も「1..*」にすることである。「顧客」と「社員」が関連していると、「顧客」のサブクラスである「技術員」のインスタンスが「顧客」のインスタンスとつながり、モデル記述と矛盾する。

被験者 F は、修正を行う前にオブジェクト図サンプルを生成している。2 回目の生成で、ある「顧客」のインスタンスが「営業員」と「技術員」の両方のインスタンスにつながっているものが生成されたことを確認している。その後、被験者 F は適切にクラスモデルの関連を修正している。修正を施した後、3 回ほどオブジェクト図サンプルを生成し、自分の施した修正が適切かどうかを確認している。

被験者 F に対し、問題 3 について尋ねたところ「おおむね修正箇所は予想できていた。システムを使った結果、その予想が正しいことを確認した」と回答している。このことから、被験者 F はモデル記述からオブジェクトモデル A を想像上に作成し、それとオブジェクトモデル B を照合していたことがうかがえる。

修正課題に成功した他の 4 例のケースにおいても、被験者 F と同様のプロセスが確認された。

5.3.3 作成課題に失敗したケース (被験者 A, 問題 3)

作成課題に失敗したケース (被験者 A, 問題 3) について述べる。被験者 A は問題 3 において、クラス図の関連を適切に修正している。しかしながら、修正課題の解答に時間を要した結果、作成課題に割くことのできる時間がわずかとなっている。被験者 A はオブジェクト図の作成をする前に、オブジェクト図サンプルを生成している。生成されたオブジェクト図サンプルのインスタンス数は 15 個であり、線が多く交差するレイアウトであった。被験者 A はレイアウトを整理せず、検証を満足に行えなっていない。その結果、作成したオブジェクト図には多重度の範囲を逸脱しているというクラス図との矛盾が存在した。

5.3.4 修正課題に失敗したケース (被験者 C, 問題 2)

修正課題に失敗したケース (被験者 C, 問題 2) について述べる。問題 2 のクラス図修正課題の正答は、「百貨店」からみた「フロア」の多重度「0..*」を「1..*」に修正することである。多重度が「0..*」の場合、「フロア」のインスタンスを持たない「百貨店」のインスタンスが生成されるため、問題 2 のモデル記述と矛盾する。

被験者 C は修正を行う前に、オブジェクト図サンプルを 4 回生成している。しかしながら、そのいずれもリンク数が 0 のパターンのオブジェクト図ではなかった。その結果、被験者 C は修正箇所が分からずそのまま問題の解答を終えている。これは、被験者 C がモデル記述から作成したオブジェクトモデルとシステムが誤っているはずのクラス

図から生成したオブジェクトモデルが一致してしまったために起きてしまったと考えられる。このような事例は、実験群にもう1例存在した。

6. 考察

本章では、実験結果を検証し、本システムの目標達成度と適用範囲について考察を行う。

6.1 UC1 (ユーザ作成のオブジェクトモデルの検証支援)の有効性

実験結果より、UC1 (ユーザ作成のオブジェクトモデルの検証支援)の有効性を考察する。正答率では、すべての問題のオブジェクト図作成課題において実験群が統制群を上回っている。5.3.1項で述べたように、UC1として筆者らが想定したプロセスによる解答が行われており、正答であった事例は11例存在した。これらを理由として、筆者らはオブジェクト図作成課題におけるUC1の有効性を主張する。

この結論の妥当性の脅威として、クラス図修正課題を正答した者のみでオブジェクト図作成課題の正答率を集計した場合、実験群と統制群で差は見られず、双方で高い正答率を示したことがある。この点について、クラス図修正課題を正答した人数はすべての問題において実験群の方が多いため、正答率に差は見られなくても正答者の数自体が多いこと、およびプロセス分析によりUC1の有効性を示す事例が11例確認できていることを根拠として妥当性を支持する。

ただし、本機能には、5.3.3項の事例で見られたように、生成されたオブジェクトモデルが複雑になるとユーザがオブジェクトモデルの照合を行えない場合があるという問題点がある。この改善案については6.3節で触れる。

6.2 UC2 (クラスモデルの誤り発見)の有効性

実験結果より、UC2 (クラスモデルの誤りの発見)の有効性を考察する。クラス図修正課題における実験群の正答率が全問を通して統制群を上回っている。被験者の解答プロセスを分析したところ5.3.2項のようなプロセスから被験者が正答していることを確認できた。筆者らがUC2として想定した解答プロセスが確認され、被験者が正答できた事例は5例あった。この5例により、正答率の上昇が見られたと考えられる。これらを理由として、筆者らはクラス図修正課題におけるUC2の有効性を主張する。

ただし、生成回数が0回や1回で、本システムを有効に利用できなかった被験者も存在したため、複数回オブジェクト図サンプルを生成したという条件付きである。もう1点、本機能には、5.3.4項の事例で見られたように、何度かオブジェクト図サンプルを生成しても、矛盾を表現するパターンが出現しないと効果的でないという問題点があ

る。この問題を改善し有効性をより高めるためには、出現パターンの制御を行う必要があり、その具体案については6.3節で触れる。

6.3 提案アルゴリズムの長所と短所

本論文で提案するアルゴリズムの長所は、学習者が低コストで様々なパターンのオブジェクト図サンプルを作成できることである。無作為選択という比較的単純なアルゴリズムによって生成されたオブジェクト図サンプルではあるが、本研究で行った実験の結果(6.1節と6.2節)は、その有効性を示している。代替案として考えられるのは、多重度から読み取れるすべてのパターンのオブジェクト図サンプルを生成する方法である。しかしながら、その方法ではクラスや関連の数が増えるに従って、生成する必要のあるオブジェクト図サンプル数が指数的に増加するという問題がある。

提案するアルゴリズムの短所は、無作為による生成のために同じパターンのオブジェクト図サンプルが出現したり、出現しないパターンが存在したりすることである。そのため、6.1節と6.2節で述べたような問題点が残っている。

UC1における問題を解決する方法として、生成されたオブジェクト図サンプルのレイアウトを制御する方法がある。これにより、ある程度のクラス数や関連数のクラス図に対して、生成されたオブジェクト図サンプルが複雑になることを抑制することができる。

UC2における問題を解決する方法は2つある。1つは、パターン選択を制御することで、同じパターンのオブジェクト図サンプルを出現しないようにすることである。少なくとも同一のパターンの出力を抑制するなどの工夫により、被験者に誤り発見の機会を増やすことができる。2つ目は、クラス図とオブジェクト図サンプルを比較し、多重度の下限や上限の場合が表現されているオブジェクト図サンプルが存在するかをチェックし、そうでなければそのパターンをユーザに提示するという機能を本システムに組み込むことである。ただし、すべてのパターンを網羅するようにしてしまうと、オブジェクト図サンプルが増大する恐れがあるため、出現していないパターンからユーザが選択してそれを生成するという方法も検討の余地がある。

6.4 本システムが有効なクラス図の複雑性

本システムの有効なクラス図の複雑性についてクラス数と関連数の観点から考察する。本論文の実験における問題に登場するクラス数は、問題1が3クラス、問題2が4クラス、問題3が5クラスである。関連数は、問題1が2つ、問題2が3つ、問題3が3つ(継承含む)である。いずれの問題も、2つ以上のクラスと関連を持つクラスは1つまたは2つであり、単純な構造をしているといえる。各問題の正答率には、クラス数と関連数との相関は見られない。

このことから、関連数が少なくクラス数が3以上で5以下の単純なクラス図においては、読解の難易度には影響がほとんどないと考えられる。しかしながら、すべての問題を通し実験群は統制群よりも高い正答率を示している。そのため、本実験で扱ったような単純な構造のクラス図であっても、学習者にとって本システムは有効であるといえる。

UC1 および UC2 においてシステムが有効的に利用されるには、様々なパターンのオブジェクト図サンプルが生成される必要がある。今回の実験において、オブジェクト図サンプルを複数生成することで、UC1 および UC2 が実験群の被験者に有効的に働いていることが確認できた。5.2 節で述べたように、実際の利用回数が少なかったことの原因として、問題が被験者にとって簡単であったという可能性があげられる。このことから、より関連数やクラス数が増加し、クラス図の可読性が低下したとき、本システムの有効性はより明確に確認されるのではないかと考えられる。今回の実験では、5 クラス以下の継承を含むモデルまでしか扱わなかったため、今後クラス数が6以上かつ関連数が多くなった場合のモデルに対しても検証が必要である。

筆者らは当初、抽象クラスの継承が含まれるモデルについては本システムが特に有効であると推測していた。これは、「抽象クラスのインスタンスが生成されない」ことが初学者にとっては理解が難しいのではないかと考えていたためである。ところが、問題3では、実験群も統制群も他の2問に比べ正答率が高かったことが確認された。したがって、今回の実験では、初学者が本システムを利用することで、抽象クラスの継承が含まれているモデルの理解が促進されたという確証は得られなかった。

しかしながら、問題3において5.3.2項で述べたような本システムを利用した際の誤り発見と検証を行うプロセスが確認された。これはUC1, UC2に該当する利用方法であり、本システムが抽象クラスの継承を含むモデルの理解に有効であったことを示している。本システムの抽象クラスの継承が含まれているモデルに対する有効性は、コンポジットパターンのような複雑性が増したものに対して、より顕著な差として確認されるのではないかと推測される。

6.5 システムの限界と今後の課題

本論文では、クラスモデルの表現するドメインの理解しやすさによる読解難易度の影響について、考慮していない。クラスモデルの表現するドメインが読解難易度に及ぼす影響についての検証と考察を行う必要がある。

現在、本システムは再帰関連や、2クラス間に2以上の関連があるクラス図には対応していない。しかしながら、これらの要素が存在するクラス図の読解は学習上重要であるとともに、初学者が躓きやすい点であると考えられる。上記のような関連へのシステム対応と評価が今後の課題である。さらに、多重度のみでなく、OCL (Object Constraint

Language) にも対応することによって、より高度なモデルに対応することも今後の課題である。

筆者らは、具体的なインスタンス名や属性値の出力もクラス図読解に有用であると考えている。しかしながら、本システムはそれらについて未対応である。現在、生成されるオブジェクト図サンプル内のインスタンスのインスタンス名は、クラス名に番号を付け加えたものである。インスタンスに具体的なインスタンス名をつけ、クラス属性を反映し属性値が出力されるように、本システムを改善することで、学習者に対し、よりクラス図の読解支援ができると思われる。

7. まとめ

オブジェクト指向モデリング教育支援を目的として、クラス図に対応したオブジェクト図を自動生成し、学習者に示すシステムを開発した。本システムでは多重度制約を考慮し、生成するインスタンスの個数およびリンク数、抽象クラスを継承しているクラスが複数存在する場合は、どのサブクラスからインスタンスを生成するかという点について、無作為に決定を行った。これにより、生成指示のたびに異なるオブジェクト図を生成し、学習者に提示するシステムを提案した。

本システムのユースケースとして、学習者が作成したオブジェクトモデルの検証支援 (UC1) とクラスモデルの多重度誤りの発見支援 (UC2) の2つを想定し、システムの評価実験を行った。大学院生を対象として、クラス数が5以下で関連数が3 (継承含む) 以下のクラス図を問題として、システムを利用する実験群7名と、利用しない統制群6名による対照実験を行った。その結果、UC1として想定された解答プロセスを被験者が行い正答した事例が11例存在した。UC1の有効性を検証するためのオブジェクト図作成課題における正答率も83.3%から100%と高い値を示し、UC1の有効性を示した。UC2として想定された解答プロセスを被験者が行い正答した事例は5例存在した。UC2の有効性を検証するためのクラス図修正課題における正答率は、すべての問題に対し実験群が統制群を上回っていたことから、UC2の有効性を示した。システムを利用することによる時間的なコストも利用しない場合と比べて0.9倍から1.1倍と無視できる程度に小さいことが分かった。

参考文献

- [1] Bezivin, J., France, R., Gogolla, M., Haugen, O., Taentzer, G. and Varro, D.: Teaching Modeling: Why, When, What?, *MODELS 2009 Workshops*, LNCS 6002, pp.55-62 (2010).
- [2] 赤山聖子, 久保秋真, 久住憲嗣, 二上貴夫, 北須賀輝明: ソフトウェアモデリング教育におけるモデル駆動開発の活用, 情報処理学会研究報告 CE113, No.5, pp.1-10 (2012).
- [3] 児玉公信: UML モデリング入門本質をとらえるシステム思考とモデリング心理学, 日経 BP 社 (2008).

[4] 中村仁昭, 小久保幹紀, 市川照久: 情報システム設計演習のためのコミュニケーションを重視した CSCL 環境の開発, 情報処理学会研究報告 IS114, pp.47-52 (2006).

[5] 安形 慶, 湯浦克彦: 学生の相互評価を用いたモデリング演習支援システムの開発, 情報処理学会研究報告 CE113, No.5, pp.1-10 (2012).

[6] Chiorean, D., Ober, I. and Petrascu, V.: Avoiding OCL specification pitfalls, *7th Educators' Symposium@MODELS 2011 - Software Modeling in Education*, pp.7-16 (2011).

[7] 城之内忠正: ゲーム作りで学ぶオブジェクト指向開発, 四日市大学環境情報論集, pp.49-64 (2006).

[8] 上原幹正, 奥平光進, 増田英孝, 笠原 宏: オブジェクトモデル作成時の過程とその支援機能の検討, 情報処理学会研究報告ソフトウェア工学研究会報告, Vol.97, No.25, pp.49-56 (1997).

[9] Kuznaiarz, L. and Staron, M.: Best Practice for Teaching UML Based Software Development, *MoDELS 2005 Workshops LNCS3844*, pp.320-332 (2006).

[10] 奥平光進, 上原幹正, 増田英孝, 笠原 宏: オブジェクトモデルを構築する際に可能な支援機能の実装, 情報処理学会第 53 回 (平成 8 年後期) 全国大会講演論文集 (1), pp.249-250 (1996).

[11] 平澤 章: UML モデリングレッスン, 日経 BP 社 (2008).

[12] Steavens, P., Pooley, R. (著), 児玉公信 (監訳): Using UML: Software Engineering with Objects and Components (邦題: オブジェクト指向とコンポーネントによるソフトウェア工学—UML を使って).

[13] Hasker, R.W. and Rowe, M.: UMLint: Identifying defects in UML diagrams, *118th Annual Conference of the American Society for Engineering Education* (2011).

[14] Hasker, R.W.: UMLGrader: An automated class diagram grader, *J. Comput. Sci. Coll.*, Vol.27, pp.47-54 (online) (2011), available from (<http://dl.acm.org/citation.cfm?id=2037151.2037163>).

[15] Auxepales, L., Py, D. and Lemeunier, T.: A Diagnosis Method that Matches Class Diagrams in a Learning Environment for Object-Oriented Modeling, *8th IEEE International Conference on Advanced Learning Technologies, ICALT '08*, pp.26-30 (2008).

[16] 杉浦啓孝, 松澤芳昭, 酒井三四郎: 凝集度と結合度に注目する OOD 学習支援システム Fourcs の提案, 情報処理学会第 73 回全国大会講演論文集, Vol.1, pp.439-440 (2011).

[17] Auer, M., Tschurtschenthaler, T. and Biffi, S.: A Flyweight UML Modelling Tool for Software Development in Heterogeneous Environments, *The 29th EUROMICRO Conference*, pp.267-272 (2003).

[18] Turner, S.A. and Manuel, A.: Perez-Quinones and Stephen H. Edwards: minimUML: A Minimalist Approach to UML Diagramming for Early Computer Science Education, *ACM Journal on Educational Resources in Computing*, Vol.5, No.4 (2005).

[19] Dranidis, D.: Evaluation of StudentUML: An Educational Tool for Consistent Modeling with UML, *Proc. Informatics Education Europe II Conference* (2007).

[20] Ramollari, E. and Dranidis, D.: StudentUML: An Educational Tool Supporting Object-Oriented Analysis and Design, *Proc. 11th Panhellenic Conference on Informatics* (2007).

[21] 金城龍弥, 樋山淳雄: インспекションコメントを利用したモデリング学習支援システム, 電子情報通信学会技術研究報告 ET (教育工学), Vol.104, No.48, pp.19-24 (2004).

[22] 金城龍弥, 樋山淳雄: インспекション成果物を利用したモデリング学習支援システムの開発と評価, 電子情報通信学会技術研究報告 KBSE (知能ソフトウェア工学),

Vol.106, No.652, pp.47-52 (2006).

[23] 竹政昭利: UML モデリング技能認定試験<入門レベル (L1)> 問題集—UML2.0 対応, 技術評論社 (2007).

付 録

A.1 問題 1

問題 以下のモデルを読み, 次の 2 つの設問に解答しなさい.

- (1) クラス図に存在する誤り (モデル記述と矛盾する多重度) を見つけて修正しなさい.
- (2) クラス図修正後のモデルについて, 典型的な場面を表現するオブジェクト図を 1 つ描きなさい.

モデル化対象 ある大学での授業履修システムである.

モデル記述

- 学生は複数の授業を履修できる.
- 履修を希望した授業は, 必ず受講できるものとする.
- 学生は, 同じ授業を再履修することも可能である.

クラス図



※履修クラスは, 学生の成績や履修年度などを記録するクラスである.

A.2 問題 2

問題 問題 1 と同様のため省略

モデル化対象 ある百貨店の店内構成のモデルである.

モデル記述

- ある百貨店の店内構成のモデルである.
- ある百貨店はいくつかのフロアで構成されている.
- 各フロアには 1 つ以上のジャンル (婦人服, 生活雑貨など) が設定されている.
- ジャンルには, それぞれに応じたテナントが 1 つ以上入っている.
- 1 つのジャンルが複数のフロアにまたがることはない.

クラス図



A.3 問題 3

問題 以下のモデルを読み, 次の 2 つの設問に解答しなさい.

- (1) クラス図に存在する誤り (モデル記述と矛盾する関連) を見つけて修正しなさい.
- (2) クラス図修正後のモデルについて, 典型的な場面を表現するオブジェクト図を 1 つ描きなさい.

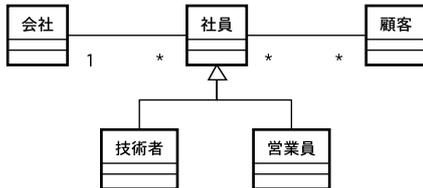
モデル化対象 ある会社の社員の社員管理のシステムで

ある。

モデル記述

- 社員は、技術者と営業員の2職種の社員がいる。
- 営業員の社員には、担当する顧客がいる。なお、技術者には担当する顧客は存在しない。
- 複数の営業員が1つの顧客を担当することもあれば、1人の社員が複数の顧客を担当することもある。

クラス図



A.4 事前試験問題

問題 以下の記述を表現する静的モデルを作成しなさい。

モデル化対象 英会話教室の会員管理システムである。

モデル記述

- (A) 会員は少なくとも1つのクラスに所属する。
- (B) 会員は途中でクラスを変更したり、複数のクラスに属することが可能である。
- (C) いずれのクラスも必ず1人のアメリカ人講師が担当する。講師は最低でも1つ以上のクラスを受け持つ。

場面記述

- (i) 会員の増田さんと田中さんは、マイケルが講師を担当するスタンダードAクラスに所属している。
- (ii) 会員の小林さんは、スミスが講師を担当するスタンダードBクラスに所属している。さらに、最近上達したため、ボビーが講師を担当するビジネスBクラスにも所属するようになった。
- (iii) 会員の熊谷さんは、講師のジョンのファンなので、彼が担当するスタンダードCクラス、海外旅行Dクラス、ビジネスEクラスのすべてに所属している。



早川 勝

2010年静岡大学情報学部卒業。2012年静岡大学大学院情報学研究科修士課程修了。現在、(株)本田技術研究所在職中。在学中は、オブジェクト指向モデリングの教育手法の研究に従事。



野沢 光太郎 (学生会員)

2011年静岡大学情報学部卒業。現在、同大学大学院修士課程。UMLによるオブジェクト指向モデリングの教育に関する研究に従事。



松澤 芳昭 (正会員)

2000年慶應義塾大学環境情報学部卒業。2002年慶應義塾大学大学院政策・メディア研究科修士課程修了。2007年同後期博士課程単位取得退学、2008年より静岡大学情報学部特任助教。現在、同学部助教。博士(政策・メディア)。オブジェクト指向技術を応用したソフトウェアの設計と開発、情報教育、情報システム技術者育成の研究に従事。日本教育工学会、情報システム学会各会員。



酒井 三四郎 (正会員)

1984年静岡大学大学院電子科学研究科博士後期課程修了。学習院大学、新潟産業大学、静岡大学工学部を経て、1998年静岡大学情報学部助教授。現在、同学部教授。工学博士。ソフトウェア開発支援環境、プログラミング学習支援環境、遠隔学習、協調学習に関する研究・開発に従事。電子情報通信学会、教育システム情報学会各会員。