

クラス図-オブジェクト図間一貫性診断システムの 設計と実用モデルへの適用実験

野沢 光太郎¹ 松澤 芳昭² 酒井 三四郎²

概要: 設計者が一貫性と明瞭性のある構造モデルを作成できるようになることを目的に、クラス図-オブジェクト図間の一貫性診断システムを開発した。初学者用に開発したシステムを基にして、実用モデルへ対応するため、診断範囲を継承、再帰関連、2クラス間に2本以上の関連がある(多重関連)、があるモデルに拡張した。実用モデルとして、(A) ファウラーの責任関係パターン、(B) 本システム自身、(C)CHARMに基づく生産管理システム、の3つのモデルを対象に適用実験を行った。その結果、(1) 実用規模のモデルに対しても効果的な診断が行えた、(2) 診断によりクラス図の修正が促進された、(3) 多くの改変を経たモデルは一貫性と明瞭性の劣化が大きい、ことがわかった。

Demonstration of Multiplicity Doctor in Practical Situations: Effects of a Class Diagram and Object Diagram Clarity and Inconsistency Diagnostic System

Abstract: We have developed a system that diagnoses clarity and consistency between class diagrams and object diagrams written by UML for professional designers. Although the system called “Multiplicity Doctor” has been developed for beginners in educational environments, we expanded constraints of the model the system can diagnose by enhancing inheritance, recursive relations, and multiple relations between two classes. This paper demonstrates that the system could be applied to three example applications: (A) Fowler’s accountability pattern, (B) The system itself, and (C) Production management system based on CHARM model. As a result, we found the following three conclusions: (1) The system effectively performed for complex and large models used in practical situations, (2) The diagnosis results promoted model improvement for professionals, and (3) Large inconsistencies were found in models with long change histories.

1. はじめに

90年代にUML(Unified Modeling Language)が登場し、オブジェクト指向に基づく開発方法論の標準として定まった。UMLでは表記法の標準化に専念した。その結果、現在ではソフトウェアの開発において、開発者や利用者間の意思疎通のために使われるようになった[1]。分析・設計者が自分の意図を正しく他の開発者や利用者へ伝えるためには、一貫性と明瞭性のあるモデルを作成する必要がある。

本研究で議論の対象とするのは、構造モデル(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、他の振る舞いや相互作用モデルの基礎

となる重要なモデルである。モデリングの経験が浅い初学者は、クラス図を作成する時に多重度を誤って付けたり、クラスとインスタンスの関係を把握できずクラスとインスタンスが混在するような図を作成してしまう事がある。モデリングの成果物には一意の正解というものが無いため、初学者が自分だけで学習を行うことが難しい。

そこで筆者らはオブジェクト図に着目したクラス図読解支援を考えた。オブジェクト図は、クラス図を具体的に表現したもので、インスタンスによる構造を記述したものである。学習者は、クラス図と対応するオブジェクト図を作成することで、具体的なレベルでのモデル読解が可能になる。そのための支援システムとして、クラス図-オブジェクト図の一貫性診断システム「MultiplicityDoctor」を開発した[2]。

筆者らは、本システムは初学者だけでなく、実際の業務

¹ 静岡大学大学院情報学研究科
Graduate School of Informatics, Shizuoka University
² 静岡大学情報学部
Faculty of Informatics, Shizuoka University

にクラス図とオブジェクト図を用いるオブジェクト指向モデリング熟練者に対しても、有用であると考えた。モデル変更が多く行われる開発現場では、モデルの一貫性と明瞭性の保持が難しくなると考えたためである。そこで、筆者らは本システムを実用モデルに対応させるために、診断範囲を継承、再帰関連、2クラス間に2本以上関連がある(以後、多重関連と呼ぶ)、があるモデルに拡張した。

本研究では、設計者が一貫性と明瞭性のある構造モデルを作成できるようになることを目的とし、クラス図とオブジェクト図間の一貫性を診断するシステムを提案する。本システムを用いて、「ファウラーの責任関係パターン」、「本システム自身」、「CHARMに基づいた生産管理システム」の3つの実用モデルを対象に適用実験を行い、本システムの有用性を示す。

2. 関連研究

本研究は、オブジェクト図を利用する点が特徴である。オブジェクト図はクラス図と異なり、文献によって重要度が異なり、扱われないものもある。我々の先行論文 [3] ではこの事情について、教育的観点からレビューを行っている。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、一般的とはいえない状況と考えられる。

専門書においては、ファウラーのアナリシスパターン [4] では一部説明にオブジェクト図 (UML ではない) が扱われている。アナリシスパターンの基礎となっているオデルの文献 [5] では、オブジェクトの説明が豊富に扱われているが、インスタンスを丸粒で扱うベン図のような図や、人の顔などのイラストでオブジェクトを表現するオブジェクト図で説明が行われる。同様の図について、シュレイヤ・メラ法のスターは「スナップショット図」と呼んでいる [6]。プロフェッショナルがクラス図の検証のためにオブジェクト図を使う方法は児玉 [1] によって提案されている。

オブジェクト図以外の UML モデルの一貫性保持支援ツールは多くのものが提案されている。Lucas らは UML モデルの一貫性保持支援ツールについて体系的にレビューを行っている [7]。レビューしている 42 の論文のうち、約 50% はクラス図とシーケンス図、ステートチャート図といった静的なモデルと動的なモデルの間の一貫性の保持支援をするものである。本研究で扱うオブジェクト図間の一貫性を保つための支援をするものは、1 件のみ [8] であり、この Haussmann らの研究でも、クラス図の多重度に注目していない。その他、クラス図とコミュニケーション図の一貫性保持支援ツールの提案が 3 件ある [9], [10], [11]。しかしながら、そのいずれもコミュニケーション図の動的側面しか扱っていない。

Ramollari らは、教育向けの UML モデリングツール

「StudentUML」を開発し、その評価を行っている [12][13]。学習者が簡単に使える事を目標としている。StudentUML はクラス図とシーケンス図の一貫性をチェックする。矛盾の自動修正の機能がある。上記の研究はモデル間の一貫性に着目した点で、本研究と同様である。しかしながら、本研究では一貫性だけでなく、明瞭性を診断することで学習者の作成するモデルの質の向上を目指している。

久保田らはインスタンスモデルベースのクラス図検証としてインスタンス図導出システム (IDIS: Instance-Diagram Induction System) を構築している [14]。IDIS は、クラス図の制約条件下で作成可能な範囲でインスタンス図の作成を支援する。IDIS では、クラス図と矛盾しているインスタンス図が発生することなく、クラス図の検証を行える事が利点である。IDIS の問題として、チェックを行うツールと違い、生成するインスタンスをユーザが選択してインスタンス図を拡張するため、「必ず 1 のみ存在する」という場合など、対応できない多重度制約がある。

一般的にオブジェクト指向モデリングによる設計に利用される CASE (Computer Aided Software Engineering) ツールに搭載されている一貫性保持支援機能について述べる。Change Vision 社の「Astah*」では、クラス図とシーケンス図間で不整合検出機能が提供されている。これは、ライフラインのベースとなるクラスの存在やメソッドの存在、クラス間の関係、可視性についての不整合を検出する。

Sparx Systems 社の「Enterprise Architect」では、クラス図とシーケンス図間の一貫性をチェックする「シーケンス図チェックアドイン」が提供されている。これは、シーケンス図内の要素がクラス図と対応がとれているかどうかを検証する。ステートマシン図と状態遷移表を連携させることで、ステートマシン図内の考慮漏れや抜けを排除するための検証支援も行っている。

このように、クラス図とシーケンス図間の一貫性保持支援機能はほとんどの CASE ツールで提供されているが、静的モデル間の一貫性保持支援機能を提供する CASE ツールは無く、多重度に関する一貫性保持支援は行われていない。

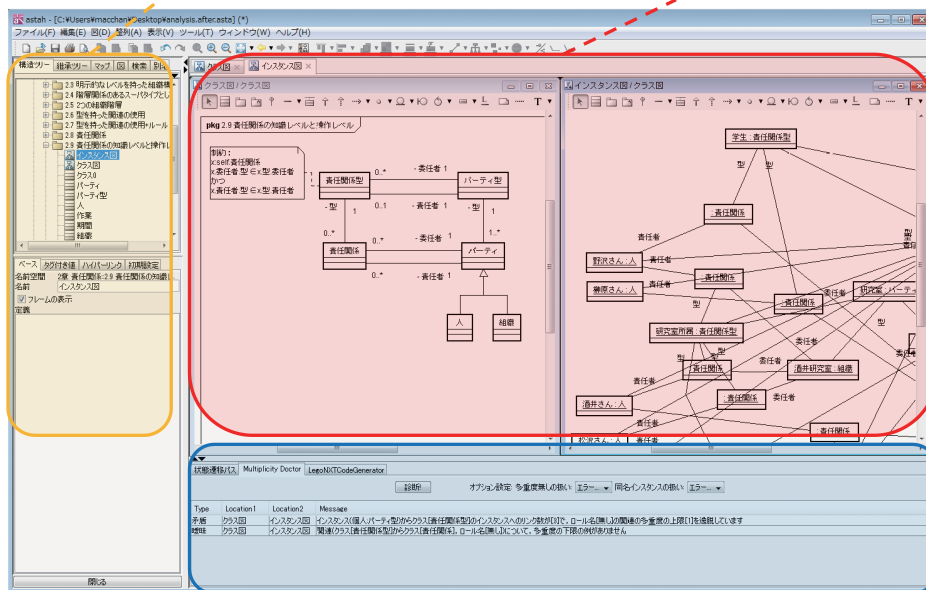
3. 提案システムの仕様

3.1 システムの目的

筆者らは、一貫性と明瞭性の 2 軸からモデルの品質について考慮すべきだと考えた。一貫性と明瞭性の有無によるモデルの品質を図 1 に示す。一貫性の無いモデルは、「誤りのあるモデル」である。一貫性が無いと、設計者の意図しているものとモデルが実際に表現するものに差異が生じる。一貫性があっても、明瞭性が無いモデルは「無意味なモデル」である。明瞭性とは、モデルの表記に曖昧な部分が無く、モデルの意図が明確であることである。例えば、全ての関連の多重度が「*対*(多対多)」の場合、どのようにインスタンスがリンクしていくのかわからず、設計者の

(1) プロジェクトビュー

(2) 図ビュー



(3) Multiplicity Doctorビュー

図 2 本システムの画面構成 (プロジェクトビューと図ビューは Astah*と同様である)

Type	Location1	Location2	Message
矛盾	クラス図	インスタンス図	インスタンス(個人/パーティ型)からクラス[責任関係型]のインスタンスへのリンク数が[3]で、ロール名[無し]の関連の多重度の上限[1]を逸脱しています
曖昧	クラス図	インスタンス図	関連(クラス[責任関係型]からクラス[責任関係], ロール名[無し])について、多重度の下限の例がありません
曖昧	クラス図	インスタンス図	クラス[責任関係]のインスタンスからクラス[パーティ]の派生クラス[パーティ]のインスタンスに繋がるリンク例がありません
曖昧	クラス図	インスタンス図	クラス[責任関係]のインスタンスからクラス[パーティ]の派生クラス[パーティ]のインスタンスに繋がるリンク例がありません
曖昧	クラス図	インスタンス図	関連(クラス[パーティ型]からクラス[パーティ], ロール名[無し])について、多重度の下限の例がありません
曖昧	クラス図	インスタンス図	クラス[パーティ型]のインスタンスからクラス[パーティ]の派生クラス[パーティ]のインスタンスに繋がるリンク例がありません

図 3 診断結果の出力例

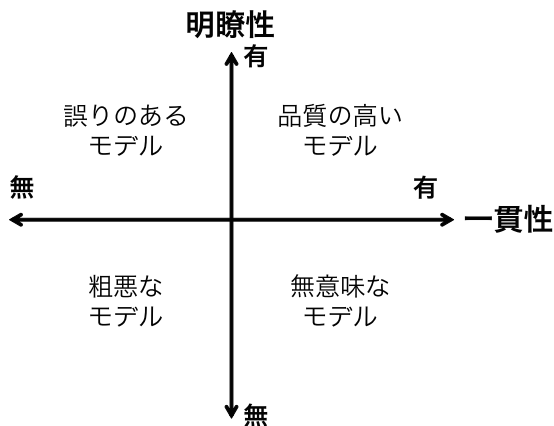


図 1 モデルの品質

意図を他者に正確に伝える事が難しくなってしまう。モデルに一貫性と明瞭性の双方がある時、「品質の高いモデル」の前提条件を満たしているという。

本システムの目的は、モデル設計者が作成したクラス図-オブジェクト図が「品質の高いモデル」の前提条件を満たせるように、支援することである。

3.2 システムの概要

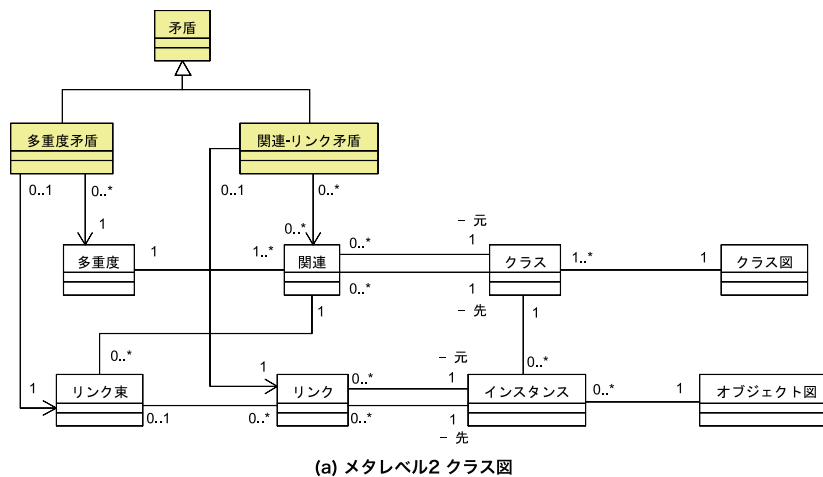
本システムは作成されたクラス図とオブジェクト図を比較し、図間の一貫性と明瞭性を診断し、その結果をユーザ

に提示するものである。本システムのユーザインタフェースを図 2 に示す。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ている。図 2 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。

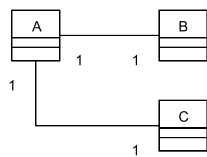
- (1) プロジェクトビュー. 同一プロジェクト内に保存されている図やモデルがツリー上に表示される。
- (2) 図ビュー. Astah*を利用して作成されるクラス図やオブジェクト図はここに表示される。
- (3) MultiplicityDoctor ビュー. 本研究で実装した箇所であり、このビューの上部中央にあるボタンが、クラス図とオブジェクト図間の一貫性と明瞭性の診断を実行するボタンである (以下診断ボタンとする)。診断結果は MultiplicityDoctor ビューの下部に表示される。

診断結果の出力例を図 3 に示す。診断結果の表示エリアは以下の構成で成り立っている。

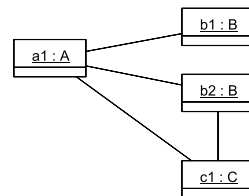
- Type: 「矛盾」、「曖昧」、「前提条件未充足」のいずれの問題かを表示する。
- Location1: 問題の発生しているクラス図を表示する。
- Location2: 問題の発生しているオブジェクト図を表示する。



(a) メタレベル2 クラス図



(b) メタレベル1 クラス図



(c) メタレベル1 オブジェクト図

図 4 矛盾の概念モデル

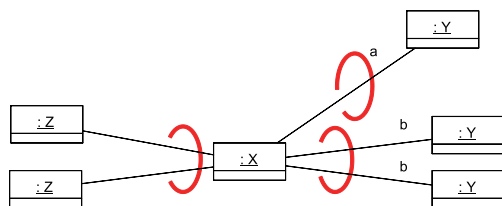
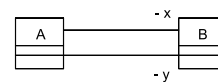
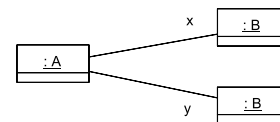


図 5 リンク束の例



(a) 二重関連のある2クラス



(b) リンクの関連端のロール名による区別

図 6 多重関連の扱い方

- Message: 問題の内容を表示する。
表示されたメッセージをクリックすることで、図ビューに Location1 と Location2 の図を表示する。

3.3 システムによる診断項目

システムによる診断項目を本システムの概念モデルを使って説明する。

補足として、本システムにおいて扱う関連とリンクは単方向のものである (例として、クラス A とクラス B の間に関連があった時、その関連を A → B の関連と B → A の関連の 2 つの関連としてモデルを構築する)。

3.3.1 一貫性診断

一貫性の診断では、以下の「矛盾」がクラス図-オブジェクト図間にあるか診断する。括弧内は 4 章で使用される簡略記号である。

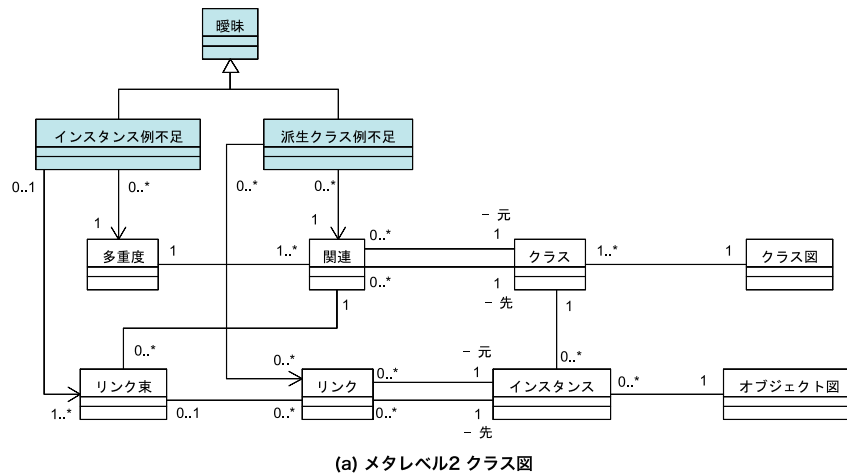
- 多重度矛盾 (I-Mul)
- 関連リンク矛盾 (I-RL)

図 4 の (a) に、本システムにおける「矛盾」の概念モデルを示す。モデル構築では、クラス、関連、インスタンス、リンクのモデルを構築した後、リンク束というモデルを構

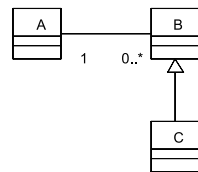
築する。リンク束とは、1 つのインスタンスが特定のクラスのインスタンスに繋がっているリンクをまとめたものである。リンク束の例を図 5 に示す。

多重関連が存在する時、関連を区別するために、図 6 に示すように、関連端のロール名とリンク端の名前によって区別し、リンク束を構築する。

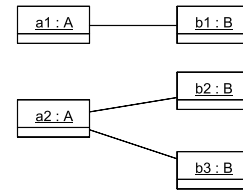
「多重度矛盾」は関連の多重度とリンク束を比較した時に、リンク束のリンク数が多重度の上限下限を逸脱している状態である。まとめられたリンクの数と、リンク束に対応する関連の多重度を比較する。図 4 の (b), (c) に、「多重度矛盾」が発生しているメタレベル 1 のクラス図とオブジェクト図の例を示す。この例では、クラス「A」から見てクラス「B」の多重度は「1」とクラス図では表記されている。オブジェクト図ではクラス「A」のインスタンス a1 からクラス「B」のインスタンスへのリンク束のリンク数は 2 であり、「1」(すなわち「1..1」) という範囲を逸脱している。



(a) メタレベル2 クラス図



(b) メタレベル1 クラス図



(c) メタレベル1 オブジェクト図

図 7 曖昧の概念モデル

「関連-リンク矛盾」は、リンクとクラス図上の関連全てと比較した時、リンクに対応する関連が存在しない状態である。図 4 は「関連-リンク矛盾」の例でもある。この例では、オブジェクト図内のクラス「B」のインスタンス b2 とクラス「C」のインスタンス c1 のリンクについて、クラス図には対応するクラス「B」と「C」の間に関連が無い。

3.3.2 明瞭性診断

明瞭性の診断では、2種類の「曖昧」がクラス図-オブジェクト図間にあるか診断する。括弧内は4章で使用される簡略記号である。

- インスタンス例不足 (A-Ex)
- 派生クラス例不足 (A-Dr)

図 7 の (a) に、本システムにおける「曖昧」の概念モデルを示す。図 7 の (b), (c) に、曖昧が発生しているメタレベル1のクラス図とオブジェクト図の例を示す。

「インスタンス例不足」は、ある関連に対応するリンク束と多重度の上限と下限を比較した時、多重度の上限又は下限と一致するリンク数のリンク束が無い状態である。多重度の上限が「*」だった場合は下限より多いリンク数のリンク束が存在していない時、「インスタンス例不足」である。図 7 の (b), (c) に、「インスタンス例不足」が発生しているメタレベル1のクラス図とオブジェクト図の例を示す。この例では、クラス「A」から見たクラス「B」への多重度は「*(0..*)」となっている。オブジェクト図では、クラス「A」のインスタンスからクラス「B」のインスタンスへのリンク束のリンク数がそれぞれ1つと2つ場合の例しかなく、下限である0の場合の例が無いため、「インスタンス

例不足」である。

「派生クラス例不足」は、関連先のクラスがサブクラスを持つような関連と、オブジェクト図の全てのリンクを比較した時、リンク先のインスタンスのベースクラスとして登場していない関連先のクラスまたはそのサブクラス（これらを派生クラスと呼ぶ）が存在している状態である。図 7 は「派生クラス例不足」の例でもある。この例では、クラス図において、クラス「B」はクラス「C」のスーパークラスである。オブジェクト図では、クラス「A」のインスタンスからクラス「C」のインスタンスにリンクがある例は無く、「派生クラス例不足」が発生している。

3.3.3 前提条件診断

前提条件の診断は、「矛盾」と「曖昧」を診断する以前に行う診断であり、(1) 関連に多重度がつけられていない「多重度不明」、(2) クラス間の関連が2つ以上ある時、ロール名が重複している「ロール名重複」、(3) インスタンスに関する不備がある「インスタンス不備」を診断する。

「インスタンス不備」とは、インスタンスにベースクラスとなるクラスが決められていない、抽象クラスのインスタンスがある、ベースクラスがクラス図に存在していない、1つのオブジェクト図内に同じインスタンス名で、同じベースクラスのインスタンスが2つ以上存在している、という問題が発生していることである。

なお、前提条件の未充足は4章において、「DNQ」の記号で表わされる。

表 1 課題の規模

	課題 A	課題 B1	課題 B2	課題 C
クラス数	6	11	11	49
関連数	12	28	28	50
インスタンス数	23	65	57	320
リンク数	80	206	158	730

表 2 課題 A. 責任関係パターンの診断結果

	1	2	3	4	5	6	計
I-Mul	12	9	1	0	0	0	12
I-RL	6	0	0	0	0	0	6
A-Ex	7	7	2	2	0	0	7
A-Dr	4	6	3	3	3	0	6
DNQ	0	0	0	0	0	0	0

表 3 課題 B1. 本システム矛盾モデルの診断結果

	1	2	3	4	5	6	7	8	9	10	計
I-Mul	-	5	4	3	0	0	3	1	0	0	8
I-RL	-	2	0	0	0	0	0	0	0	0	2
A-Ex	-	17	18	19	18	15	0	1	1	0	20
A-Dr	-	0	0	0	0	0	0	0	0	0	0
DNQ	1	0	0	0	0	0	0	0	0	0	1

4. 実用モデルへの適用実験

三つの実用モデルに対し、本システムの適用実験を行った。本章ではその方法と結果について述べる。

4.1 3つの課題と規模

本研究で選択した課題は以下の3つである。

- (A) アナリシスパターン [4] における責任関係パターン
- (B) 本システム自身
- (C) 少量多品種型生産管理システムの一般モデル CHARM [15], [16] に基づく生産管理システム

このうち、「B. 本システム自身」に関しては、矛盾と曖昧の2つについてのモデルを課題とした。それぞれの問題を「B1. 矛盾」、「B2. 曖昧」とする。

課題の規模を表 1 に示す。課題 C と比較して課題 A, B のクラス数は少数であるが、いずれも継承、再帰関連、多重関連を含む課題である。

4.2 課題 A. 責任関係パターン

アナリシスパターン [4] の1つである責任関係パターンのクラス図^{*1}に対して、本論文の筆者の1名(松澤)が手動でオブジェクト図を作成し、本システムを利用して診断を行った。利用したクラス図、作成したオブジェクト図の一部を付録の図 A-1^{*2}、図 A-2 に付す。

本実験では、課題に対してクラス図とオブジェクト図間で矛盾と曖昧が出現しなくなるまで診断と修正を繰り返し、診断回数と各診断における矛盾と曖昧の出現数を記録した。その結果を表 2 に示す。この表では、行やエラーの出た種類を示し、列は診断回数を示し、各セルは本システムによる指摘の件数を示す。例えば、「1列」の「I-Mul

*1 同書内の「図 2.9 責任関係の知識レベルと操作レベル」を一部省略したもの

*2 図 A-1 には多重度の誤りがあるが、その発見の過程を示すため、あえてそのまま掲載している。

行」は「1回目の診断出現した多重度矛盾の件数」を示す。「計」欄は修正過程で出現した指摘の合計を示すが、以前の診断から引き続き指摘されているものはカウントしない。つまり、単純に行を合計したものが計欄とはならない。

結果は、矛盾が18件(多重度矛盾12件、関連-リンク矛盾6件)であり、曖昧が11件(インスタンス例不足7件、派生クラス例不足4件)であった。クラス図またはオブジェクト図の修正と診断を繰り返し、6回目の診断で矛盾と曖昧の出現数が0になっている。この過程で、筆者はパーティパターンを複製したクラス図のミスを発見し、修正を行っている。クラス図の修正箇所は1か所であった。出現した矛盾の総数は18件(多重度矛盾12件、関連-リンク矛盾6件)、曖昧の総数は13件(インスタンス例不足7件、派生クラス例不足6件)であった。

4.3 課題 B. 本システム自身

3.3節に示した本システムのクラス図(図4, 図7)に対して、本論文の筆者の1名(野沢)が手動でオブジェクト図を作成し、本システムを利用して診断を行った。作成したオブジェクト図の一部を付録(図 A-3 と図 A-4)に付す。

図 A-3 は、図4で示した本システムの矛盾のメタレベル1の例をメタレベル2のオブジェクト図で示したものである。図 A-4 は、図7で示した本システムの曖昧のメタレベル1の例をメタレベル2のオブジェクト図で示したものである。

本実験では、矛盾、曖昧の各課題に対してクラス図とオブジェクト図間で矛盾と曖昧が出現しなくなるまで診断と修正を繰り返し、診断回数と各診断における矛盾と曖昧の出現数を記録した。

4.3.1 課題 B1. 矛盾モデル

矛盾モデルの診断結果を表 3 に示す。表の読み方は表 2 と同様である。矛盾が7件(多重度矛盾2件、関連-リンク矛盾5件)、曖昧が17件(インスタンス例不足17件、派生クラス例不足0件)であった。矛盾と曖昧の出現数が0になるまでの診断回数は10回であった。出現した矛盾の総数は10件(多重度矛盾8件、関連-リンク矛盾2件)、曖昧の総数は20件(インスタンス例不足20件、派生クラス例不足0件)であった。修正の過程において、クラス図の修正箇所は2か所であった。

4.3.2 課題 B2. 曖昧モデル

曖昧モデルの診断結果を表 4 に示す。表の読み方は表 2

表 4 課題 B2. 本システム曖昧モデルの診断結果

	1	2	3	4	5	計
I-Mul	0	2	0	0	0	2
I-RL	0	0	0	0	0	0
A-Ex	15	5	5	2	0	15
A-Dr	4	0	0	0	0	4
DNQ	0	0	0	0	0	0

表 5 課題 C. CHARM に基づく生産管理システムの診断結果

	1	2
I-Mul	-	142
I-RL	-	452
A-Ex	-	41
A-Dr	-	99
DNQ	75	0

と同様である。矛盾が 0 件、曖昧が 19 件 (インスタンス例不足 15 件, 派生クラス例不足 4 件) であった。矛盾と曖昧の出現数が 0 になるまでの診断回数は 5 回であった。出現した矛盾の総数は 2 件 (多重度矛盾 2 件, 関連-リンク矛盾 0 件), 曖昧の総数は 19 件 (インスタンス例不足 15 件, 派生クラス例不足 4 件) であった。修正の過程において, クラス図の修正箇所は 4 か所であった。

4.4 課題 C. CHARM に基づく生産管理システム

CHARM に基づいて設計された生産管理システムのモデルに対して, 本システムを適用した。対象モデルのクラス図は, クラス数 49, 関連数 50 のクラス図であり, 200 回以上の改訂が行われている大規模で改訂量の多いモデルである。対象のクラス図の一部を抜粋し, 難読化処理を施したものを付録の図 A-5 に付す。実用モデルへの適用に際し, 診断において同名同クラスのインスタンスを許可し, 多重度のつけられていないものは「0..1」と解釈した。

本実験は現在進行中であり, 結果は研究会にて補足発表の予定である。本稿では診断の途中結果を図 5 に示す。初回診断では, インスタンスのベースクラスがクラス図に存在していない「前提条件未充足」の問題が 75 件発見されている。これらに関しては, インスタンスのベースクラスを現在のクラス図にあるものを置き換え, 置き換えることが可能なクラスが無い場合はインスタンスそのものを消去する修正が必要であった。2 回目の診断の結果は, 矛盾が 594 件 (多重度矛盾 142 件, 関連-リンク矛盾 452 件), 曖昧が 140 件 (インスタンス例不足 41 件, 派生クラス例不足 99 件) であった。

5. 考察

課題 A, B, C の結果より, 本システムによって実用規模のモデルに対して, 一貫性と明瞭性の効果的な診断が行えたと言える。課題 A, B では本システムの診断結果により, 筆者らが実際に矛盾の修正と曖昧に対する検証を行え

ていた事も, この結論を支持する。

課題 A, B より, 本システムの診断によってクラス図の修正が促進されたことが言える。課題 A ではオブジェクト図を作成し, 診断において矛盾を指摘された結果, 多重度のミスを一ヶ所発見し修正した。課題 B では, オブジェクト図を作成し, 矛盾と曖昧を指摘され検証を行った結果, クラス図を計 6 ヶ所修正したことより, この結論を支持する。

課題 C の診断結果より, 実用規模のモデルにおいては改変数が多くなると一貫性と明瞭性の劣化が大きくなると言える。課題 A, 課題 B で扱ったモデルは, 最新版のクラス図からオブジェクト図を作成したため, 一貫性と明瞭性の劣化はほとんどないと言える。課題 C ではオブジェクト図を作成してから何度も改変を経た結果, 他 2 つの課題と比べ, 20 倍以上の矛盾と曖昧が発生していた。

6. おわりに

設計者が一貫性と明瞭性のある静的モデルの作成を目的として, クラス図-オブジェクト図間の一貫性と明瞭性を診断するシステムを開発した。本システムの診断範囲を継承, 再帰関連, 2 クラス間に 2 本以上の関連がある (多重関連), があるモデルに対応させた。「ファウラーの責任関係パターン」, 「本システム自身の」, 「CHARM に基づく生産管理システム」の 3 つのモデルを対象に適用実験を行った。実験の結果, クラス数 49 で関連数 50 のクラス図とインスタンス数 320 でリンク数 730 の規模の実用モデルに対しても効果的な診断が行えた。診断により, クラス図の修正が促進された例が 7 例確認された。200 回以上の改変を経たモデルは矛盾が 594 件, 曖昧が 140 件と一貫性と明瞭性が劣化している事がわかった。

謝辞 本研究に際してアドバイスをいただいた児玉公信さんに感謝します。

参考文献

- [1] 児玉公信: UML モデリングの本質 良いモデルを作るための知識と実践, 日経 BP 社 (2004).
- [2] 野沢光太郎, 松澤芳昭, 酒井三四郎: オブジェクト指向モデリング学習のためのクラス図-オブジェクト図間の一貫性・明瞭性診断システムの提案と評価, 情報教育シンポジウム (SSS2012), pp. 249-256 (2012).
- [3] 早川 勝, 野沢光太郎, 松澤芳昭, 酒井三四郎: 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, 情報処理学会論文誌, Vol. 54, No. 1 (2013).
- [4] Fowler, M.: Analysis Patterns: Reusable Object Models (邦題: アナリシスパターン), Addison-Wesley (1996).
- [5] Odell, J. J.: *Advanced Object-Oriented Analysis & Design Using UML*, Cambridge University Press, New York, NY (1998).
- [6] Starr, L.: *How to Build Shlaer-Mellor Object Models*, Prentice Hall (1996).
- [7] Lucas, F. J., Molina, F. and Toval, A.: A systematic

review of UML model consistency management, *Information and software Technology*, Vol. 51, pp. 1631-1645 (2009).

- [8] Hausmann, J., Heckel, R. and Sauer, S.: Extended model relations with graphical consistency conditions, *Proceedings UML 2002 Workshop on Consistency Problems in UML-based Software Development*, pp. 61-74 (2002).
- [9] Lucas, F. and Toval, A.: A precise approach for the analysis of the UML models consistency, BP-UML' 05: 1st International Workshop on Best Practices of UML, in 24th International Conference on Conceptual Modeling (ER 2005), Springer (2005).
- [10] Paige, R., Brooke, P. and Ostroff, J.: Metamodel-based model conformance and multiview consistency checking, *ACM Transaction on Software Engineering*, Vol. 16, No. 3 (2007).
- [11] Laleau, R. and Polack, F.: Using formal metamodels to check consistency of functional views in information systems specification, *Inf. Softw. Technol.*, Vol. 50, No. 7-8, pp. 797-814 (2008).
- [12] Ramollari, E. and Dranidis, D.: StudentUML:An Educational Tool Supporting Object-Oriented Analysis and Design, *Proceeding of the 11th Panhellenic Conference on Informatics* (2007).
- [13] Dranidis, D.: Evaluation of StudentUML:An Educational Tool for Consistent Modeling with UML, *Proceeding of the Informatics Education Europe II Conference* (2007).
- [14] 久保田 吉彦幹正, 増田英孝, 笠原 宏: 対話的にオブジェクトモデルを確認するためのインスタンス図導出システムの作成, 情報処理学会全国大会講演論文集第 57 回平成 10 年後期 (1), pp. 203-204 (1998).
- [15] 児玉公信, 水野忠則: 少量多品種型生産管理システムの一般モデル CHARM の提案, 情報処理学会論文誌, Vol. 49, No. 2, pp. 902-909 (2008).
- [16] 児玉公信: 計画・実行システムの一般モデルー生産管理システムと金融業務システムの共通性ー, 情報処理学会研究報告 (IS109-4) (2009).

付 録

A.1 課題 A. 責任関係パターンのモデル

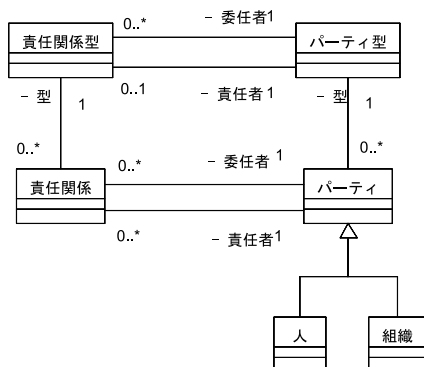


図 A.1 課題 A. 責任関係パターンのクラス図 (誤りがあるもの)

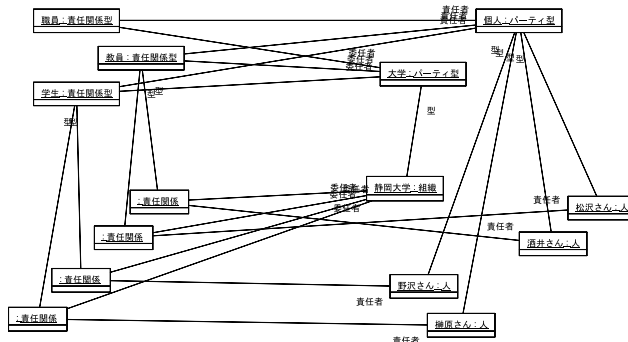


図 A.2 課題 A. 責任関係パターンオブジェクト図

A.2 課題 B. 本システム自身のモデル

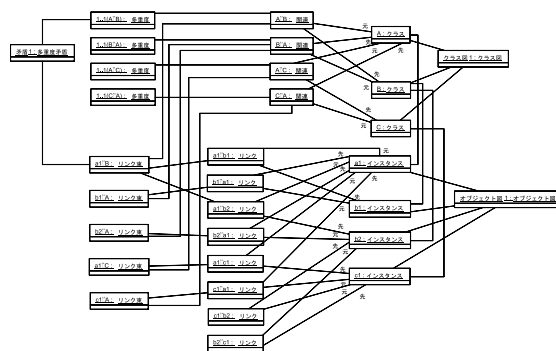


図 A.3 課題 B1. 矛盾例の M2 オブジェクト図

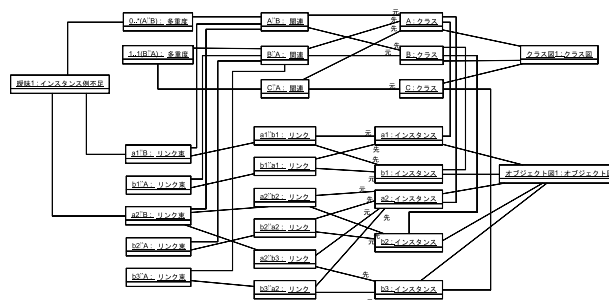


図 A.4 課題 B2. 曖昧例の M2 オブジェクト図

A.3 課題 C. CHARM に基づく生産管理システムのモデル

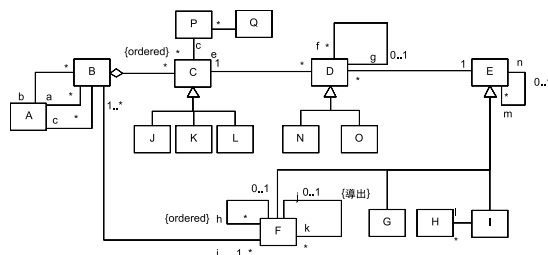


図 A.5 課題 C. CHARM に基づく生産管理システムのクラス図 (抜粋, 難読化処理済)