

第 3 章

クラス継承木の進化的特徴

3.1 概要

クラス継承木の進化過程について SystemA を調査した結果、クラス継承木には 1 メソッド当たりの行数の値を保持し続ける傾向があることを発見した。この傾向は、たとえば、あるクラスが継承木に追加された場合、そのクラスは継承木が持つ 1 メソッド当たりの行数に近い値をもつように設計されるという意味を持ち、たとえ初期の値が継承木の持つ値に近くなかったとしても、再設計によって、継承木が持つ値に近い値を持つように進化するという意味を持つ。本章では、クラスの 1 メソッド当たりの行数を C 値 (Characteristic Value) と呼ぶ。また、クラス継承木の C 値とは、クラス継承木に属するクラスの C 値の平均値を指す。本研究では、クラス継承木の進化的特徴を捉えるために、C 値に関するクラス継承木の進化の仮説を提起し、他の 2 システムを用いて仮説の検証を行った。

仮説を検証した結果、C 値には次の 3 つの特徴があることがわかった。

- C 値はクラス継承木に固有の値である。
- C 値はシステムが成長しても安定しており、クラスは、自分が属するクラス継承木の C 値を持つように進化する傾向がある。
- C 値は開発者に依存せずに決まる。

本章の構成は次のとおりである。第 3.2 節では SystemA で調査した結果を説

表 3.1: SystemA におけるクラスの性質を表す計測値の相関分析結果 (相関係数)

	CNOM	CLOC	DIT	NOC	NIV
CNOM	1.00				
CLOC	0.88	1.00			
DIT	-0.01	-0.18	1.00		
NOC	0.15	0.01	-0.28	1.00	
NIV	0.86	0.62	0.16	0.15	1.00

明し、仮説を提起する。第 3.3 節では、他の 2 システムを用いた仮説の検証結果を示す。最後の節では、仮説が示す設計上の意味について議論する。

3.2 予備調査結果

3.2.1 計測値の相関分析結果

SystemA のクラスレベルで計測した値に対して相関分析を行った結果、表 3.1 に示すように、メソッド数 (CNOM) とクラス行数 (CLOC)、CNOM とクラスのインスタンス変数の数 (NIV) の間に強い正の相関があった。

散布図を書いたところ、特にクラスの CNOM に対する CLOC の散布図で両者の間に比例関係を確認することができた。図 3.1 には、横軸に CNOM をとり、縦軸に CLOC をとったクラスの散布図を示す。図 3.1 の左上は、SystemA の全版の全クラスの値を示した散布図である。他の図には、クラスを継承木ごとに分類して散布図を示した。散布図に示したクラスの継承木は、SystemA を構成する次の主なクラス継承木である。

- 編集クラス木：シミュレーションに必要な初期値を与える利用者インタフェースのクラス群。
- 表示クラス木：シミュレーションの設備を表すアイコンなど、設備を視覚的に組み立てる利用者インタフェースを提供するクラス群。

表 3.2: 継承木ごとの CLOC と CNOM の間の相関係数

分析対象	全クラス	表示クラス木	編集クラス木	計算クラス木
相関係数	0.871	0.969	0.951	0.994

- 計算クラス木：シミュレーションの計算エンジンを提供するクラス群。

ここで、クラスの 1 メソッド当たりの行数の値を C 値 (Characteristic Value) と定義する。すなわち、クラス i について、その行数を $CLOC_i$ 、メソッド数を $CNOM_i$ 、 C 値を CV_i とすると、次の関係が成り立つ。

$$CLOC_i = CV_i \cdot CNOM_i$$

図 3.1 の矢印は第 3 版で新たに定義された編集クラス木のクラスが、第 4 版に至る期間で変化した軌跡を表している。矢印の頭が、継承木の C 値に向かっていることから、この間に行われた設計変更がクラスの C 値を継承木の C 値へ向かわせたと考えることができる。開発者はインタビューのなかで、第 3 版のクラスの設計には違和感があり、第 4 版を開発する際に修正したと言っていた。このような調査結果は、クラスの C 値と継承木の C 値との差が、開発者にとって直観的な設計上の不適切さとして認識されていたことを表している。

表 3.2 には、全クラスと継承木ごとに求めた $CNOM$ に対する $CLOC$ の相関係数を示す。各継承木の相関係数が全クラスについて求めた相関係数より大きいことから、 C 値をクラス継承木に分割して観測することにした。表 3.3 に SystemA の各クラス継承木における C 値の基本統計量を示す。直観的にクラス継承木の C 値は継承木ごとに異なり、システムが成長しても大きな変化がないように見えるが、これを統計的手法を用いて検定する。

検定にあたり帰無仮説 $H_0(C$ 値固有) を提起し、2 組ずつ継承木を抽出して C 値の平均値について有意水準 5% の両側 t 検定を行う。

$$H_0(C$$
値固有) : 2 つのクラス継承木の C 値は等しい

表 3.4 に t 検定によって得られた有意水準 P 値を示す。 P 値が検定で用いた有意水準 5% よりも小さいことから、帰無仮説 $H_0(C$ 値固有) を棄却できる。す

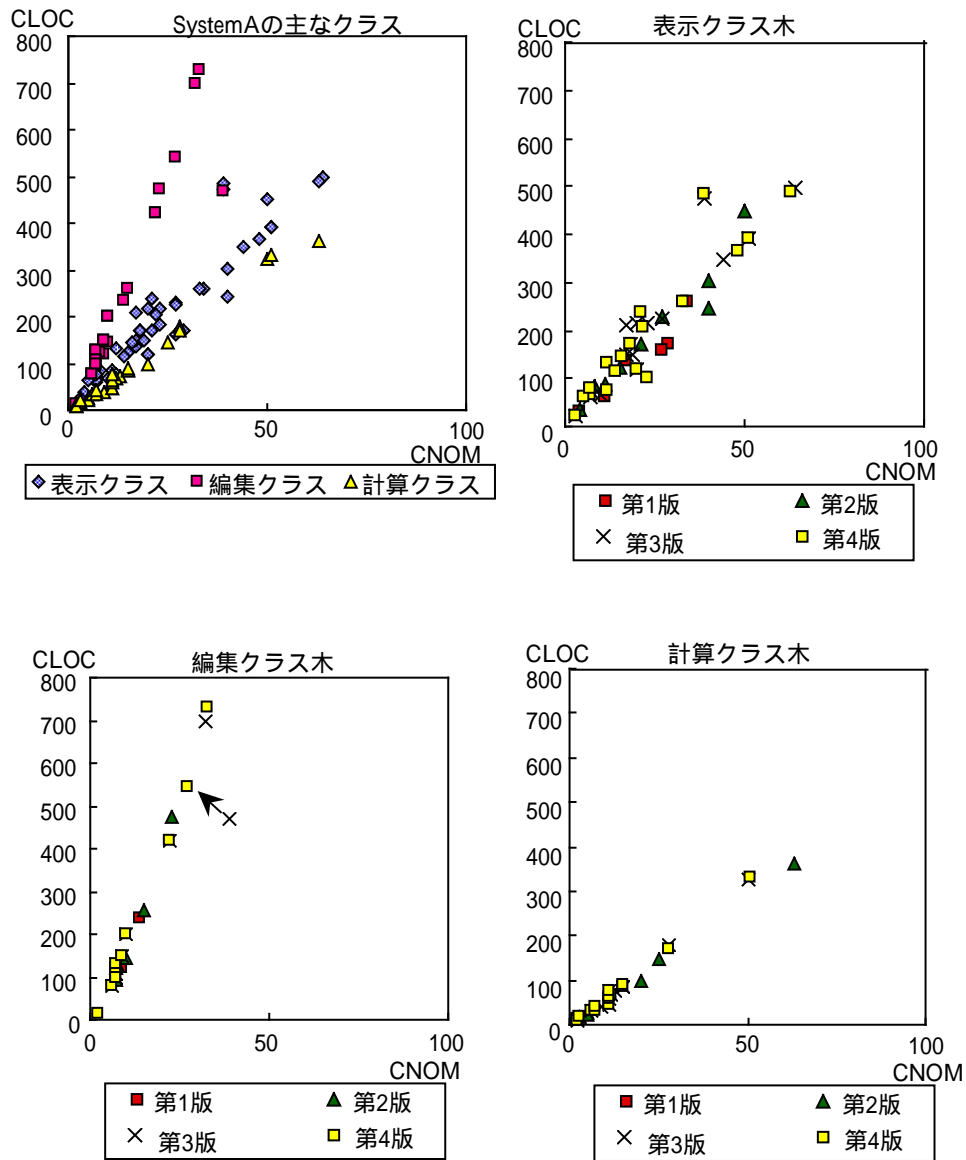


図 3.1: SystemA のクラスのメソッド数 (CNOM) に対するクラス行数 (CLOC) の散布図

表 3.3: SystemA における各継承木の C 値に関する基本統計量

表示クラス木	第 1 版	第 2 版	第 3 版	第 4 版
平均値	7.63	8.44	8.63	8.93
標準偏差	1.89	1.27	1.62	1.78
最小値	5.55	6.10	6.00	6.00
最大値	10.86	10.50	12.29	12.60
クラス数	6	8	16	18
編集クラス木	第 1 版	第 2 版	第 3 版	第 4 版
平均値	15.10	16.34	16.32	16.24
標準偏差	2.16	2.63	3.11	4.07
最小値	13.14	13.14	12.00	7.00
最大値	17.14	20.61	21.81	22.09
クラス数	4	6	11	12
計算クラス木	第 1 版	第 2 版	第 3 版	第 4 版
平均値	-	5.43	5.68	5.74
標準偏差	-	0.70	0.90	0.85
最小値	-	4.40	4.18	4.18
最大値	-	6.33	7.00	7.00
クラス数	-	6	15	15

表 3.4: SystemA における両側 t 検定によって求めた有意水準 : P 値 (表の有意水準 P 値が検定で用いた有意水準より小さいとき帰無仮説を棄却する)

	編集クラス木	表示クラス木	計算クラス木
編集クラス木	-	-	
表示クラス木	1.05E-15	-	
計算クラス木	2.22E-19	2.04E-16	-

なわち , すべてのクラス継承木の C 値が等しいとは言えないことを統計的に確認できた .

次にクラス継承木の C 値の安定性を評価するために , 各版間の平均値と分散に関する帰無仮説 H_0 (平均値安定) , H_0 (分散安定) を提起し , 有意水準 5% の両側 t 検定および両側 F 検定を実施した .

H_0 (平均値安定) : クラス継承木の C 値の平均値はシステムの成長によって変化しない

H_0 (分散安定) : クラス継承木の C 値の分散はシステムの成長によって変化しない

表 3.5 に , t 検定および F 検定によって求めた有意水準 P 値を示す . 表に示す各 P 値は検定で用いた有意水準 5% より大きいから , 帰無仮説を棄却できない . すなわち , システムが成長したとき , 各クラス継承木の平均値および分散が変化するとは言えないという結論を統計的に得ることができた .

3.2.2 クラス継承木の進化的特徴に関する仮説

SystemA において観測されたクラス継承木の進化的特徴が一般的に観測されるか否かを検証するために , 次に列挙する 3 つの仮説を検証した .

1. C 値固有仮説 : クラス継承木の C 値はクラス継承木間で差異はない .
2. C 値安定仮説 : クラス継承木の C 値は , システムが成長しても , その平均値 , および分散は変化しない .
3. C 値開発者非依存仮説 : C 値は , 開発者間で有意な差異はない .

これらの仮説を検証するために SystemB および SystemC のクラス継承木の進化的特徴を調査した .

表 3.5: SystemA の各クラス継承木ごとの C 値の平均値および分散に対する両側 t 検定と両側 F 検定より求めた有意水準：P 値

t 検定	1-2 版	2-3 版	3-4 版
表示クラス木	0.229	0.470	0.662
編集クラス木	0.459	0.989	0.963
計算クラス木		0.485	0.923
F 検定	1-2 版	2-3 版	3-4 版
表示クラス木	0.275	0.273	0.795
編集クラス木	0.793	0.746	0.405
計算クラス木	-	0.590	0.771

3.3 仮説の検証

3.3.1 SystemB の検証結果

クラス継承木の概要

次に，SystemB を構成する 4 つの主なクラス継承木を示す．仮説の検証は，これらのクラス継承木を対象として行った．

- 仲介クラス木：利用者インタフェースとデータベース上のオブジェクト間の仲介を行うクラス群．
- 一覧表操作クラス木：一覧表に表示されたオブジェクトの操作を行うクラス群．
- 永続化クラス木：データベース上の永続オブジェクトに対応するクラス群．
- 表抽出クラス木：データベース上のオブジェクトを一覧表に表示するために，データベースからオブジェクトを抽出するクラス群．

仲介クラス木と一覧表操作クラス木は共通のスーパークラスを継承し，永続化クラス木と表抽出クラス木は，最終版でクラス Object の直下に定義された．

仮説の検証

1. C 値固有仮説の検証

図 3.2 にすべてのクラスと 4 つのクラス継承木で求めた各クラスの CNOM に対する CLOC の散布図を示す。図 3.2 には、クラスの進化の軌跡を矢印で示した。また、上段の散布図に示した直線の傾きは、各クラス継承木の C 値を表す。

相関分析の結果、クラス継承木の相関係数の最大値は一覧表操作クラス木の 0.994、最小値は表抽出クラス木の 0.915 で、いずれのクラス継承木でも CNOM と CLOC の間に強い正の相関関係を観測できた。相関分析の詳細は、付録 C に示してあるので参照されたい。

表 3.6 にクラス継承木ごとの C 値に関する基本統計量を示す。表では、第 1 版から第 2 版の間で各クラス継承木の C 値が変化しているが、これは第 1 版のクラス数が少ないために発生した誤差と考えられる。

表 3.7 は 5% の有意水準の両側 t 検定で求めた各クラス継承木の P 値を示したものである。永続化クラス木と表抽出クラス木の間は P 値は 0.05 より大きいが、他のクラス継承木の間で求めた P 値は 0.05 よりも小さい。以上の結果から、帰無仮説 H_0 (C 値固有) は採択できない。すなわち、すべてのクラス継承木の C 値は等しいとは言えないことが SystemB においても統計的に検証できた。

2. C 値安定仮説の検証

仲介クラス木に定義されている各クラスの CNOM に対する CLOC の値の散布図を、図 3.2 の中段左に示す。円内の点を結んでいる矢印は、第 2 版から第 3 版へ至る期間のクラスの進化の軌跡である。この軌跡の矢印の頭がクラス継承木の C 値へ向いているのは、第 3 版でクラスの設計が変更された結果、このクラスの C 値がクラス継承木の C 値に近い値へ変化したことを表している。この現象は SystemA の編集クラス木で観測した現象と同じである。他のクラス継承木の散布図を観察しても、クラスはクラス継承木の C 値に沿って進化しているように見える。

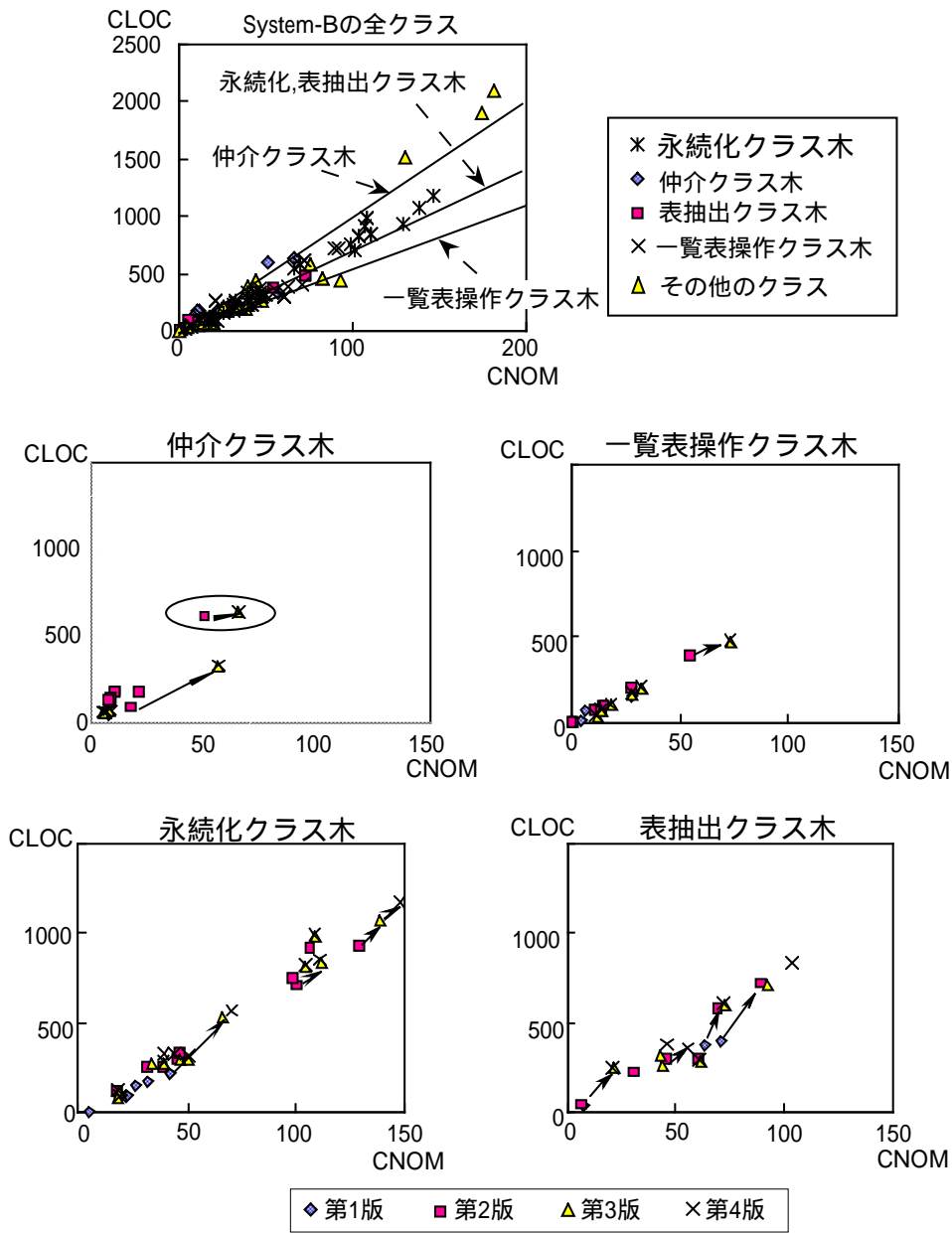


図 3.2: SystemB の CNOM に対する CLOC の散布図

表 3.6: SystemB の各クラス継承木の基本統計量

仲介クラス木	第 1 版	第 2 版	第 3 版	第 4 版
平均値	8.04	12.29	9.06	9.07
標準偏差	1.80	4.53	1.64	1.64
最小値	7.00	4.56	5.74	5.74
最大値	10.11	16.00	10.00	10.00
クラス数	3	7	6	6
一覧表操作クラス木	第 1 版	第 2 版	第 3 版	第 4 版
平均値	8.21	5.28	5.90	5.91
標準偏差	3.64	2.08	0.76	0.76
最小値	5.5	2.00	4.45	4.45
最大値	13.33	7.27	6.66	6.66
クラス数	4	11	7	7
永続化クラス木	第 1 版	第 2 版	第 3 版	第 4 版
平均値	4.97	7.08	7.37	7.49
標準偏差	1.26	0.78	1.10	1.13
最小値	2.25	6.17	5.12	4.95
最大値	6.00	8.58	9.18	9.18
クラス数	8	10	11	11
表抽出クラス木	第 1 版	第 2 版	第 3 版	第 4 版
平均値	5.77	6.80	7.67	7.77
標準偏差	0.30	1.17	2.37	2.38
最小値	5.43	4.85	4.89	4.89
最大値	6.10	8.26	12.29	12.29
クラス数	4	7	7	7

表 3.7: SystemB における両側 t 検定によって求めた有意水準 : P 値

クラス木名	仲介	一覧表操作	永続化	表抽出
仲介	-			
一覧表操作	0.000	-		
永続化	0.000	0.004	-	
表抽出	0.001	0.006	0.505	-

表 3.8: SystemB の各クラス継承木で求めた C 値の平均値および分散に対する両側 t 検定と両側 F 検定で求めた有意水準：P 値

t 検定	1-2 版	2-3 版	3-4 版
仲介クラス木	0.164	0.128	0.996
一覧表操作クラス木	0.068	0.459	0.992
永続化クラス木	0.001	0.337	0.812
表抽出クラス木	0.164	0.888	0.861
F 検定	1-2 版	2-3 版	3-4 版
仲介クラス木	0.284	0.041	0.998
一覧表操作クラス木	0.157	0.022	0.991
永続化クラス木	0.196	0.394	0.926
表抽出クラス木	0.144	0.151	0.983

C 値安定仮説を検証するために、クラス継承木ごとに求めたクラスの C 値の平均値および分散に対して、有意水準 5% の両側 t 検定と両側 F 検定を行い帰無仮説 H_0 (平均値安定) および H_0 (分散安定) を検証した。検定によって求めた有意水準 P 値を表 3.8 に示す。ほとんどのクラスについて検定で用いた有意水準 5% よりも検定結果の有意水準 P 値は大きい。永続化クラス木の第 1 版と第 2 版の間で求めた t 検定の有意水準 P 値が 5% よりも小さく、したがって平均値に有意な差異があるという結果が得られたのは、第 1 版の時点でデータベースの設計が終了していなかったため、第 1 版から第 2 版へ至る間に設計変更が起きていたためと考えられる。このような設計変更の例を除くと、帰無仮説 H_0 (平均値安定) は棄却できない。すなわち、クラス継承木の版ごとに求めたクラスの C 値の平均値は、隣接する版の間に差異があるとは言えないという結論が得られた。

また、F 検定の結果(表 3.8)では、第 2 版から第 3 版の期間における仲介クラス木と一覧表操作クラス木の P 値は 0.05 よりも小さい。一覧表操作クラス木の P 値は、第 2 版で 2.00、2.50 という分散の値を持つ 2 クラスの C 値が有意な差異を生じさせた原因であった。これらのクラスは、クラス継承木を構成しており、第 3 版では削除された寿命の短いク

ラスであった。そこで、これらのクラスを例外として一覧表操作クラス木から除いたところ、第 2 版から第 3 版の期間における P 値は 0.05 よりも大きくなった。

仲介クラス木は、図 3.2 に示したクラスの進化の軌跡でもわかるように、クラスの進化の方向はクラス継承木の C 値に向かって収束している。このように C 値の分散は新たなクラスが追加された際に一時的に大きくなることもあるが、そのような時期を例外と考えると、帰無仮説 H_0 (分散安定) を棄却することはできない。

以上の検定結果から、C 値の安定仮説を検証できた。

3.3.2 SystemC の検証結果

クラス継承木の概要

SystemC では、次の 8 つのクラス継承木に着目して、仮説を検証する。ただし、SystemC は漸進的な開発が行われなかったため、C 値安定仮説については検証することができなかった。そのかわり、C 値開発者非依存仮説を検証できた。クラス継承木の担当者と共に、それぞれのクラス継承木の概要を説明する。

- 証券領域クラス木群：証券問題領域のクラスが構成するクラス継承木で、5 つの子継承木、およびその他のクラスからなる。5 つの子継承木のうち 3 つを開発者 a が開発し、残りの 2 つのクラス継承木を開発者 b が開発した。
- 仲介クラス木群：データベース上のオブジェクトと証券領域クラスとの仲介を行うクラスが構成するクラス継承木で、開発者 c が開発した 2 つの子継承木を持つ。
- 定数表クラス木：税金などの定数表を保持するクラスが構成するクラス継承木で、開発者 d が開発した。

仮説の検証

1. C 値固有仮説の検証

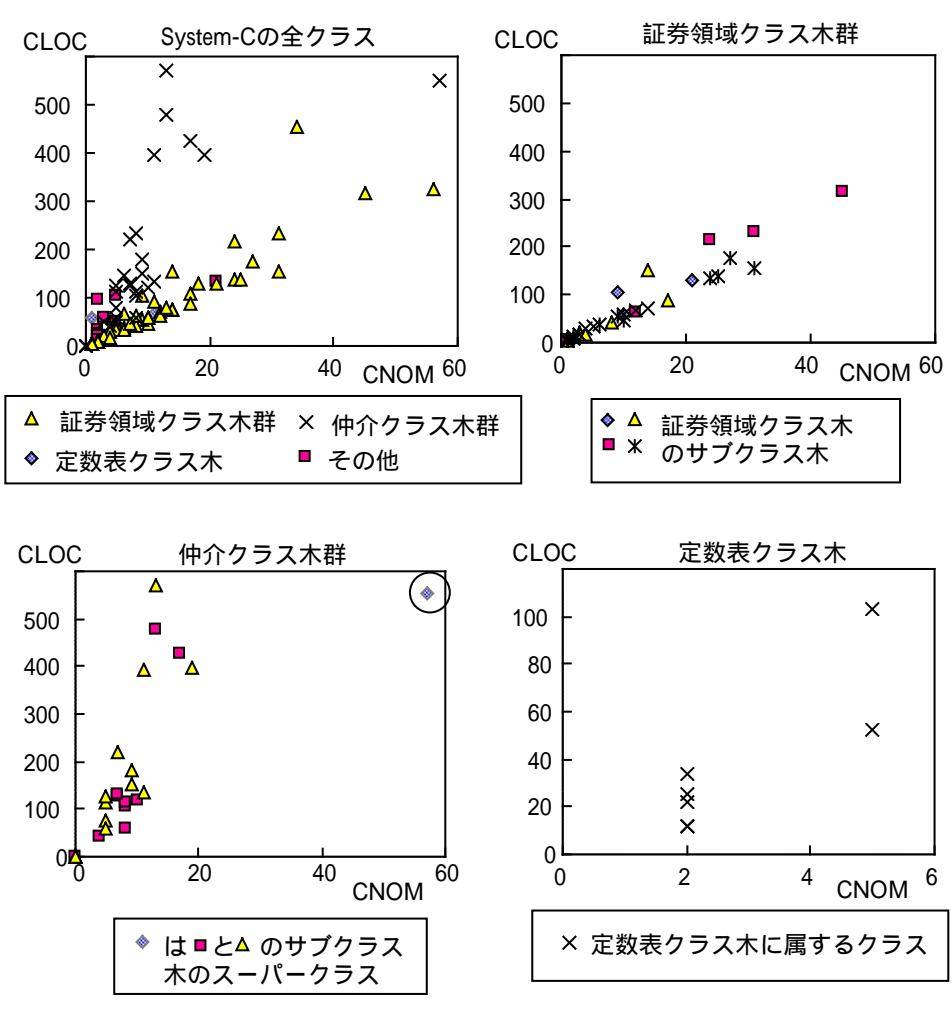


図 3.3: SystemC の CNOM に対する CLOC の散布図

表 3.9: SystemC のクラス継承木における C 値の基本統計量

クラス木名	証券領域 1	証券領域 2	証券領域 3	証券領域 4
平均値	4.87	5.17	5.99	5.58
標準偏差	0.98	1.23	2.54	1.98
最小値	4.00	4.00	4.00	4.00
最大値	6.56	7.50	10.93	9.00
クラス数	13	12	6	8
クラス木名	証券領域 5	仲介 1	仲介 2	定数表
平均値	4.90	17.30	23.28	16.44
標準偏差	2.20	8.97	10.10	13.70
最小値	4.00	7.38	11.60	6.00
最大値	11.78	36.77	43.77	48.00
クラス数	13	9	11	8

図 3.3 に、SystemC の全クラスとクラス継承木ごとに分けたクラスの CNOM に対する CLOC の値を散布図で表した。グラフから、CNOM と CLOC の間に強い正の相関関係を観測できる。相関分析の結果、全クラスにおける相関係数は 0.793、証券領域クラス木で 0.927 の最大値が得られた他、最小の定数表クラス木でも 0.561 が得られた。定数表クラス木の相関係数が小さいのは、CNOM および CLOC の値がともに小さく、値の小さい変動が相関係数に大きな影響を与えているためである。以上から SystemC の CLOC と CNOM の間にも、正の相関があることを統計的に確認できた。

これらのクラス継承木を用いて、帰無仮説 $H_0(C$ 値固有) の検定を行う。検定は、有意水準 5% の両側 t 検定で行った。表 3.9 に、SystemC の第 14 版の全クラスを対象にして、クラス継承木ごとに求めたクラスの C 値の基本統計量を示す。表 3.10 には t 検定の結果から求めた有意水準 P 値を示した。P 値から、観測に用いたクラス継承木は二つのグループに分類できることがわかる。二つのグループとは、証券領域クラス木を構成する 5 つの継承木と、仲介クラス木を構成する 2 つの継承木である。これらのグループ内のクラス継承木の C 値には有意な差を認めることはで

表 3.10: SystemC のクラス継承木のクラスを対象とした C 値に対する両側 t 検定で求めた有意水準 : P 値 (ただし, 証券 n は証券領域 n のクラス継承木を表す)

クラス木	証券 1	証券 2	証券 3	証券 4	証券 5	仲介 1	仲介 2	定数表
証券 1								
証券 2	0.510							
証券 3	0.337	0.483						
証券 4	0.369	0.610	0.752					
証券 5	0.964	0.708	0.389	0.474				
仲介 1	0.003	0.004	0.005	0.004	0.003			
仲介 2	0.000	0.000	0.000	0.000	0.000	0.178		
定数表	0.049	0.053	0.068	0.062	0.050	0.882	0.254	

きないが, グループ間のクラス継承木の C 値には, 有意な差が認められる。これは, 証券領域クラス木を構成する 5 つの子継承木は, 共通の C 値を持っており, また, 仲介クラス木に属する 2 つの子継承木も C 値を共有していることを表す。

以上の結果から, 帰無仮説 $H_0(C$ 値固有) を棄却する。すなわち, クラス継承木の C 値には, クラス継承木間で有意な差異が認められる。

帰無仮説 $H_0(C$ 値固有) を検証するにあたり, 図 3.3 の左下の図でマークをつけたクラスは, 仲介クラス木からは除いた。このクラスは 2 つの仲介クラス木の共通のスーパークラスである。スーパークラスの役割については, 次節で考察することにする。

2. C 値開発者非依存仮説の検証

SystemC では, 二人の開発者が証券領域クラス木に属するクラスを分担して開発した。そこで, 人間に依存する差異がクラス継承木の C 値へ及ぼす影響について調査し, C 値開発者非依存仮説の検証を行った。第一の開発者 a は証券領域クラス木 1, 2 および 3 を開発し, 第二の開発者 b は証券領域クラス木 4 と 5 を開発した。表 3.10 に, 帰無仮説 H_0 (開発者): 「C 値は, 開発者間で有意な差異はない」を検証するために行った

両側 t 検定の結果を示す。検定で用いる有意水準は 5% とした。検定結果は、これまでと同様、検定で得られた有意水準の P 値で示す。表 3.10 の P 値はいずれも 0.05 よりも大きいから、帰無仮説 H_0 (開発者) を棄却できない。したがって、「 C 値開発者非依存仮説： C 値は、開発者間で有意な差異はない」を検証できた。

3.4 仮説の検証結果と考察

仮説を検証した結果、 C 値には次の 3 つの特徴があることがわかった。

- C 値はクラス継承木に固有の値である。
- C 値はシステムが成長しても安定しており、クラスは、自分が属するクラス継承木の C 値を持つように進化する傾向がある。
- 開発者に依存せずに決まる。

ただし、今回行った調査では、同一開発組織の開発者による成果物を対象としているため、これらの成果物は同一のプログラミング規約のもとで開発されたものである。ただし、ここで得られた結論は、プログラミング規約の異なる開発者間でも C 値は変わらないことを意味しているわけではない。

クラス継承木の設計はこれまで、形式的に議論されることが多かったが [36]、本研究によって、クラス継承木内のクラスはクラス継承木の C 値の制約を受けながら進化する傾向があることを確認できた。この成果から、定量的にもクラス継承木の設計の妥当性を検証できることを明らかにできた。

本研究で C 値に用いた行数は単純な改行の回数であるが、行数は同一のプログラミング規約のもとではメッセージ送受信回数との関連が強い。単純に考えると、クラス継承木に分類されたクラスたちが、インスタンス変数の参照と更新だけを行うメソッドだけを持つ受動的なクラスであるとしたら、これらのクラスの C 値は一樣に小さくなる。また、継承木に分類されたクラスたちが、1 回のメッセージ受信をトリガーとして多くのメッセージを送信するクラスであるとしたら、これらのクラスの C 値は共通に大きくなると考えられる。したがって、 C 値はクラスの役割の粒度を表すとみなすことができる。

C値がクラス継承木ごとに固有の値となる原因は、継承の設計指針に依存すると思われる。継承が他のクラスの振る舞いを再利用するために使われる [59] 場合、開発者には、継承するライブラリクラスを変更する権限を与えられていないことが多い。そのため、継承を用いた再利用は、新たなクラスの分類方針を設計する作業となる。新たにアプリケーションで定義された継承木では、第2章で示したように、継承関係にあるクラスや兄弟クラスが再設計の対象となる現象を多く観測した。ここで、新たに追加されたクラスのC値がクラス継承木のC値に近づくのは、開発者が継承木に集められたクラスを同程度に複雑な振る舞いを持つようにクラスの役割の粒度を調整しているためと考えられる。

たとえば、メッセージのシグネチャが共通で内部のメソッドがクラスごとに異なる場合、スーパークラスに定義されるメソッドの行数は最小となるが、サブクラスに定義されるメソッドの行数は多くなる。この例では、シグネチャだけがスーパークラスに定義され、メソッドがサブクラスに定義されるからである。各サブクラスに定義されるメソッドはスーパークラスから継承した構造とシグネチャに基づいて実装されるため、それらのメソッドの規模が同程度とならない理由は考えられない。つまり、サブクラスどうしのC値は同程度となる可能性は高い。図 3.3の左下の図はこのような設計の例である。この図でマークをつけたスーパークラスのC値は小さく、そのサブクラスである仲介クラス木のクラスのC値はスーパークラスのC値よりも一様に大きく、同程度の値を示している。

また、シグネチャが共通で手続きもほとんど同様の内容である場合、スーパークラスには、共通の手続き部分だけをまとめたメソッドを定義し、サブクラスにはメソッドを再定義してサブクラス固有の手続き部分だけを定義する。この場合、スーパークラスの役割の粒度は大きくなるが、サブクラスの役割の粒度は小さくなるため、やはりサブクラスどうしのC値は同程度となるはずである。

ここで得られた研究の成果によって、クラス継承木のC値を、継承木に新たに追加されるクラスのC値の定量的な設計妥当性確認の指針として応用できることが明らかとなった。今後は、実際にC値を開発支援として使うための評価を進めていきたい。

