

階層状態遷移図における記述品質の 評価手法の提案

松井 聡¹ 徳本 修¹ 中島 毅^{2,a)}

受付日 2019年6月30日, 採録日 2019年11月29日

概要: 階層状態遷移図を用いた振舞い設計では、いかに適切に状態の階層化を行うかが課題である。適切な階層化が行えない場合は階層的な状態遷移図が複雑になり、可読性や生産性を低下させるという問題が生じるためである。本論文では、この問題を解決するために、階層状態遷移図の記述品質を評価する手法を提案する。この手法は、階層状態遷移図における状態と状態遷移の2つの視点から、階層性を評価する複雑度を定義している。単一の階層状態遷移図に対して、その記述品質を評価できるようにするため、視点別に規模測定量を定義し、それらを用いて当該複雑度を正規化した測定量を定義している。3つの分野と規模の異なる問題に対する記述実験により提案手法の評価を行い、問題による測定値の偏りが小さいこと、設計の良否を判定する基準値を設定できる可能性があることを示した。

キーワード: 階層状態遷移図, UML, 記述品質, 可読性, 保守性, 複雑度

A Method for Evaluating Descriptive Quality in Hierarchical State Transition Diagrams

SOICHI MATSUI¹ SYUICHI TOKUMOTO¹ TSUYOSHI NAKAJIMA^{2,a)}

Received: June 30, 2019, Accepted: November 29, 2019

Abstract: In the behavioral design using hierarchical state transition diagrams, it is crucially important to choose a good hierarchy of states. A bad hierarchy would make the diagrams complicated, resulting in the problem of reducing their readability and productivity. This paper proposes a method to evaluate the quality of hierarchical state transition diagrams to help choose a better hierarchy. This method defines a set of complexity measures to evaluate hierarchical state transition diagrams from two viewpoints: state and state transition. To make it possible to evaluate the quality of a single diagram, scale measures for the two viewpoints are defined, which can be used to normalize the complexity measures. Three experiments to describe three different problems were conducted for evaluating the proposed method, and their results show that the bias of measured values due to the difference of problems is relatively small, and that it is possible to set reference values to judge whether the quality of the diagrams are acceptable or not.

Keywords: statecharts, UML, descriptive quality, understandability, maintainability, complexity

1. はじめに

組込みソフトウェア開発は、短工期・高品質の傾向が強

まっている。状態遷移図は、システムの動的な振舞いを記載するのに適しており、状態遷移図または表を利用したプロジェクトの割合は約8割、モデルベース開発において自動コード生成で使用された上位言語である割合は約2割となっている [1]。

状態遷移図の利点は、数学的な背景を持つことにより、形式検証や、プログラムコードやテストケースの生成が可能でありながら、図的な記述であるため、人による理解が比較的容易であることにある。

¹ 三菱電機株式会社
Mitsubishi Electric Corporation, Chiyoda, Tokyo 100-8310, Japan

² 芝浦工業大学
Shibaura Institute of Technology, Koto, Tokyo 135-8548, Japan

a) tsnaka@shibaura-it.ac.jp

オリジナルの状態遷移図は、状態の間に階層構造を持っていない（以下、単純状態遷移図と呼ぶ）。記述対象のシステムが複雑になると、単純状態遷移図では、記述すべき状態数や状態遷移数が増え、現実的な設計で利用することは困難である。この問題を解決するために、Harel は、状態に階層構造を導入した状態遷移図 Statecharts を考案した [2]。

Statecharts は実用性が高いため、多くの開発現場で使われ、UML (Unified Modeling Languages) や SysML (System Modeling Languages) において、ステートマシン図として採用されている。本論文では、以降、Statecharts およびステートマシン図を総称して、階層状態遷移図と呼ぶこととする。

階層状態遷移図を用いた振舞い設計では、いかに適切に状態の階層化を行うかが課題である。適切な階層化が行えない場合は階層的な状態遷移図が複雑になり、可読性や生産性を低下させるという問題が生じる。

本論文は、この問題を解決するために階層状態遷移図の記述品質を評価する手法を提案する。この手法は、階層状態遷移図における状態と状態遷移の2つの視点から、階層性を評価する測定量を定義する。この測定量は、単一の階層状態遷移図に対して、その記述品質を評価できるようにするため、視点別に規模測定量を定義し正規化を行う工夫を行っている。

本論文は、2章で階層状態遷移図とそれを用いた設計上の課題、3章で階層状態遷移図の記述品質を評価する従来手法とその課題について述べる。4章で、階層性を評価する手法に対する要件とアプローチ、5章で提案手法とその定義、および設計根拠などを示す。6章では、実験による評価を行い、7章で関連研究、8章で結論と今後の課題を示す。

2. 階層状態遷移図と記述上の課題

2.1 階層状態遷移図

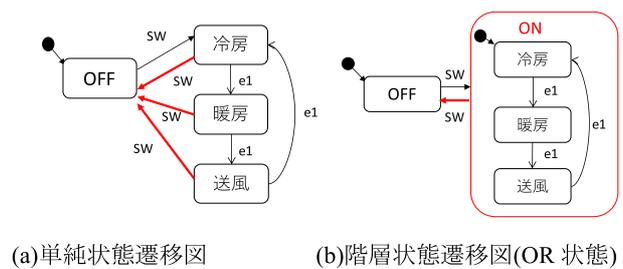
階層状態遷移図は、単純な状態遷移図に階層性と並列性を持たせることでコンパクトな記述が可能になる記述法である。ここで、ある状態の排他的な分割 (OR 分割) によりできる状態を OR 状態、ある状態の並列的な分割 (AND 分割) によりできる状態を AND 状態と呼ぶ。

(1) 階層性

階層性とは、複数の OR 状態を、これを含む共通の状態により表現することである。この共通の状態を抽象状態と呼ぶ。抽象状態もまた OR 状態である。

図 1 のエアコンの例を使用して説明する。

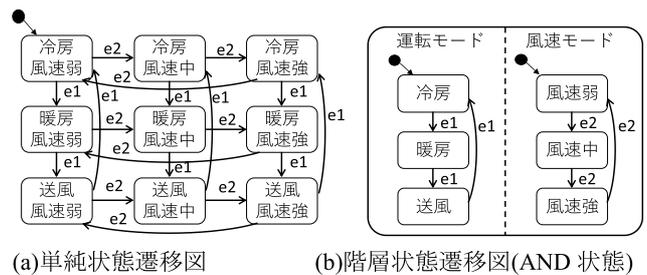
図 1 において、エアコンの運転モードについて、(a) は単純状態遷移図、(b) は「ON」という抽象状態を導入し階層状態遷移図で記述したものであり、仕様の的には同じものを表現する。ON は、冷房、暖房、送風の3つの OR 状態に共通の状態となる。(a) の冷房、暖房、送風から OFF 状



(a)単純状態遷移図 (b)階層状態遷移図(OR 状態)

図 1 エアコンの運転モードの記述例

Fig. 1 Statechart for operation modes in air conditioners.



(a)単純状態遷移図 (b)階層状態遷移図(AND 状態)

図 2 エアコンの運転モードと風速モードの記述例

Fig. 2 Statechart for operation modes and wind speed modes in air conditioners.

態への3つの状態遷移は、状態遷移図に階層性を持たせることで、(b)のようにON状態からの1つの状態遷移に集約することができる。つまり、状態遷移の数を減らすことができる。

また、ON状態は、複数の状態をまとめることで、抽象状態の内外を分け、モジュール化の効果を得ることもできる。

(2) 並列性

並列性とは、独立した複数の条件因子を、並列した状態として表現することである。

図 2 はエアコンの例を使って AND 状態を説明したものである。図 2 (b) において、エアコンは状態値「冷房」、「暖房」、「送風」をとりうる「運転モード」と、状態値「弱」、「中」、「強」をとりうる「風速モード」という2つの独立した状態変数を持つものと考えられる。階層状態遷移図では、この状況を、1つの状態を点線で分離した2つの領域 (AND 状態) を作り、それぞれに各状態変数に対する状態遷移図を記述することで表現できる。

状態遷移図に AND 状態を持たせる拡張によって、独立した機器の並列動作をそのままの形で記述可能になる。これによって、単純状態遷移図では図 2 (a) のように状態の掛け算でしか表現できず、状態数も状態遷移数も組合せ的に増えてしまうのに対して、図 2 (b) の階層状態遷移図は、状態数をコンパクトに保ちながら、状態遷移数を減少させるといったメリットがある。

2.2 階層状態遷移図の記述品質および課題

階層状態遷移図により、規模が大きく複雑なシステムをコンパクトに記述することができ、厳密な意味を持った可

読性の高い要求仕様や設計仕様を作成することが可能になる。これにより後工程の実装やテストを円滑に進めることができる。これにより、要求仕様や設計仕様の保守性の向上を図ることができる。

一方で、階層性と並列性の導入により、同じ問題を記述したとしても、階層性や並列性をどこにどれだけ記述すればよいかは設計者に任されることになり、設計記述の自由度が増加する。設計記述の自由度が増せば、設計者のスキルや経験によって同じ仕様書の状態遷移図でも異なる図が作成され、記述品質にも差ができてしまう。

ここで階層状態遷移図の記述品質を以下の2つの特性として定義する。

- **モジュール性**：仕様を独立した部分に分割でき、部分間の影響が疎であること。階層性の導入により抽象状態の上位レベルと下位レベルが分離され、並列性の導入により並行状態が分離され、かつ部分間の依存関係が少ないこと。モジュール性は、記述の可読性を高め、開発における実装ミスを減らし、設計レビューの効果と効率を上げるとともに、仕様レベルでの再利用性の向上を図ることが期待できる。
- **実装容易性**：状態遷移の発火条件が整理され、実装すべきアクション数が少なくなることにより、仕様が実装しやすいこと。実装容易性は、実装する機能数が減り実装コード規模を減らす効果を上げるとともに、可読性やテスト容易性の向上を図ることが期待できる。

階層状態遷移図の記述品質を高くすることにより、後工程の実装やテストを円滑に進めることができる。要求仕様や設計仕様の保守性の向上を図ることができる。

図3および図4に示すエアコンの運転モードの例を使って、階層状態遷移図の記述品質を考える。図1と図2の例では、e1 イベントにより冷房、暖房、送風を順番に移動する仕様であったが、図3の単純状態遷移図では、cool, heat, blow イベントにより直接任意の運転モードにも移動できる仕様となっている。

図4(a), (b)は、図3の単純状態遷移図と同じ振舞いを持つ階層状態遷移図の2つの例である。図4(a)は悪い例である。この例では、「圧縮機作動中」という抽象状態を導入している。単純状態遷移図に比較して、状態遷移数が1本減少している（実装容易性の向上）が、抽象状態をまたぐ状態遷移が5本できてしまっている。すなわち、この抽象状態の導入により、その内部の構成要素に対する変更が外部の構成要素に影響を直接与えるようになる（モジュール性の阻害）。さらに見た目の複雑さも増してしまっている。

これに対して、図4(b)は良い例である。この例では、状態遷移数が8本減り実装すべき機能数が減ることで、実装容易性が増し、抽象状態の内と外で状態遷移が分離でき、モジュール性を十分に発揮できているのが分かる。

このように、抽象状態の導入方法の違いにより記述の良

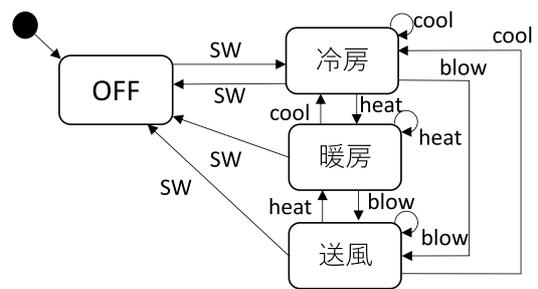


図3 対象とする単純状態遷移図

Fig. 3 A simple state transition diagram.

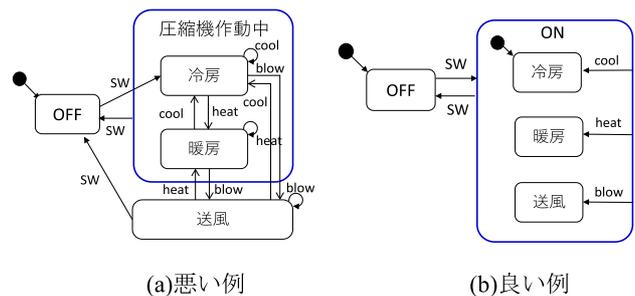


図4 図3に対応する階層状態遷移図の記述例

Fig. 4 Statechart corresponding to Fig. 3.

し悪しが大きく異なってしまいます。記述品質の悪い状態遷移図は後の工程に悪い影響を及ぼす可能性が高い。そのため、階層状態遷移図を作成した時点で、その記述品質を評価し改善点を指し示す手段があれば、設計の時点で修正ができ、後工程や保守に悪影響が及ばなくなると期待できる。

3. 従来研究

ここでは、階層状態遷移図の記述品質の測定に関する従来研究とその実用性の課題について述べる。

3.1 基本測定量

階層状態遷移図の品質を測定するための測定量として、階層状態遷移図のノード（状態）とエッジ（状態遷移）に関して、以下に示す基本測定量を用いるものがある [3]。

- **状態関連**：状態数、抽象状態数、末端状態数
- **状態遷移関連**：状態遷移数、状態またぎ数

これらの測定量は、図中に現れる要素を直接測ることができるため、測りやすく、かつ分かりやすい。

図4(a), (b)の例に対して基本測定量を測った結果を表1に示す。ここで、状態またぎ数は、状態遷移が通過するOR状態の境界をまたいだのべ総回数とする。

表1において、ノード（状態）に関する測定量は、すべて(a)と(b)で同じ数値となったのに対して、エッジ（状態遷移）に関する測定量である状態遷移数および状態またぎ数は、(a)の方が多くなっている。

この比較によって、同じ振舞いを持つ仕様で状態遷移数が少なく状態またぎ数も少ない(b)の階層状態遷移図は、

表 1 図 4 のノードとエッジに関する基本測定量

Table 1 Basic measurements related to Fig. 4.

基本測定量		(a) 悪い例	(b) 良い例
状態関連	状態数	5	5
	抽象状態数	1	1
	末端状態数	4	4
状態遷移関連	状態遷移数	12	5
	状態またぎ数	5	0

階層性をうまく利用できていることが分かる。

このように、これらの基本測定量を用いた階層状態遷移図の記述品質の評価は、複数の記述間の比較による相対評価となる。これらの基本測定量は、問題の複雑さに依存するので、複数の比較対象がないとよし悪しが判断できないためである。

こうした比較による記述品質の評価は、リファクタリングした際その効果を定量化することなどに向いているが、単独の階層状態遷移図の記述品質の良否を評価する目的には利用できない。一般的に、同じ問題の仕様を複数人の設計者が独立して同時に記述する状況は実験以外に想定しにくいためである。

3.2 サイクロマティック複雑度の応用

3.2.1 サイクロマティック複雑度とは

サイクロマティック複雑度は、グラフの複雑度を測定するための測定量である。サイクロマティック複雑度 CC は、単一グラフに対して以下の式により求めることができる。

$$CC = E - N + 2$$

ここで、 E : エッジの数

N : ノードの数

サイクロマティック複雑度は、ソフトウェア工学の分野ではプログラムコードの制御構造の複雑度として利用されてきた [4]。以下がそのやり方である。

- ① ノードは分岐点 (IF 文など)、エッジは制御の流れとして CC を計算する。
- ② 関数などの単位ごとに、 CC を評価する。

CC は、問題自体の複雑さに強く依存する。コード規模 (行数) も同様であるが、単純に設定を行うだけで分岐のないプログラムは、コード規模が増えても CC は増えない。分岐の数はテストのしやすさに関係するので、コード規模より CC で評価の方が問題の複雑さをよく表すものと考えられる。

CC を「1 関数あたり」で評価することで、関数の品質を表す指標として利用できるようになる。たとえば、関数あたり CC は 6 以下であればよしとされ、10 以上は複雑すぎると判断される。このように、 CC は、プログラム関

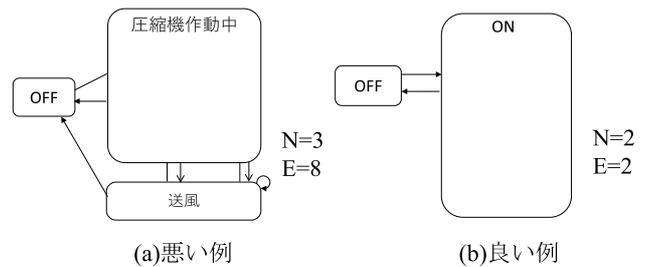


図 5 図 4 から第 1 階層レベルの状態を取り出したもの
Fig. 5 Statechart at the first hierarchy level extracted from Fig. 4.

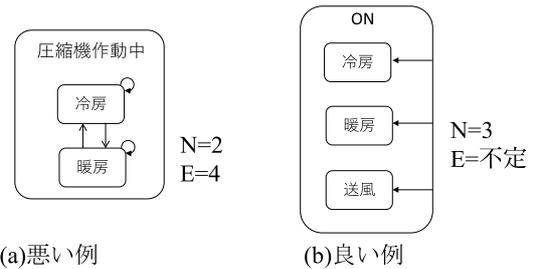


図 6 図 4 から第 2 階層レベルの状態を取り出したもの
Fig. 6 Statechart at the second hierarchy level extracted from Fig. 4.

数の単体テスト (およびレビュー) のしやすさを表す指標として広く利用されるようになっていく。

(2) 階層状態遷移図への応用

Hall は、サイクロマティック複雑度を、以下のように階層状態遷移図へ応用することを提案している [4]。

- ① ノードは状態、エッジは状態遷移に対応付ける。
- ② 階層レベルごとに CC を評価する。

サイクロマティック複雑度は、状態遷移図を状態と遷移のみを見るグラフとしての複雑度を表すものである。階層レベルを切り口に、グラフを対象に評価値を作っている。

このやり方で、図 4(a) と (b) の階層状態遷移図の CC を求めてみる。まず図 4 について、最上位階層レベルの状態遷移図を図 5 に、第 2 階層レベルの状態遷移図を図 6 に示す。

図 5 で、状態階層をまたぐ状態遷移は、一番上の階層レベルでカウントされるものとしている。初期状態やヒストリはノードとしてカウントせず、初期状態やヒストリからの状態遷移もエッジとしてカウントしない。

表 2 は図 4 に対するサイクロマティック複雑度である。表 2 において、第 1 階層レベルの値はそれぞれ 7 と 2 となり、図のよし悪しを相対的に表す測定量として利用できそうである。また、階層レベルを対象として評価しているので、規模に依存せず、絶対値として評価できるのではないかと期待できる。意味としては、各階層の CC 値を大きくしないという意味で、階層性 (抽象状態の導入) を小まめにバランス良く取り入れるための指標となりうる。

しかし、実際に適用しようとする以下の問題がある。

表 2 図 4 に対するサイクロマティック複雑度
Table 2 Cyclomatic complexity for Fig. 4.

階層レベル	サイクロマティック複雑度 CC	
	(a)	(b)
第 1 階層レベル	7	2
第 2 階層レベル	4	測定不可

- 抽象状態を多く導入することにより，確実に各階層の複雑度は低下する．無駄に導入してもペナルティはないため，抽象状態の導入の仕方自体のよし悪しを評価できない．
- モジュール性に悪影響のある状態またぎ数については，評価できていない．
- 第 2 階層レベル（表 2(b)）のように，抽象状態からその下位の OR 状態への遷移はよく使われる手法であるが，この場合の計算方法が設定されていない．
- AND 状態に対する考慮がない．

4. 評価手法に対する要求と技術的課題

4.1 評価手法に対する要件

従来研究の課題に基づき，階層状態遷移図の記述品質を評価する手法に対して，以下の 4 つの要件を設定した．

- ① AND 状態を含む階層状態遷移図に対して，階層性のよし悪しを判断できる．
- ② 単独の仕様に対して記述品質を評価できる．
- ③ 少数の測定値セットで記述品質を評価できる．
- ④ 測定値に基づき仕様に対する改善点が分析できる．

4.2 研究アプローチと技術的課題

階層性の品質評価をするための指標を開発するために，上記 4 つの要件を満たすことが期待できる，以下の 2 つの測定量により評価をすることとした．

$$Es = \text{状態の階層性評価/状態規模} \quad (1)$$

$$Et = \text{状態遷移の階層性評価/状態遷移規模} \quad (2)$$

要件①を満たすために，「状態の階層性評価」および「状態遷移の階層性評価」という独立した 2 つの測定量を定義する．要件②を満たすためには，これらの測定量を規模によらず絶対値として評価できるようにする必要がある．このために，「状態規模」および「状態遷移規模」という 2 つの規模測定量を定義し，それらを使って先の 2 つの階層性評価測定量を正規化する方法をとる．これにより， Es と Et という 2 つの独立した測定量を得る．

Es と Et が 2 個の測定量セットであることは，要件③を満たす．測定値を散布図としてプロットすることで，2 次元平面上の位置で評価することも可能になる．また， Es は状態の階層性の測定量であり，この測定値が悪いときには，表 1 の状態に関する基本測定量を見て，どこが悪いの

かを分析しやすい． Et についても同様である．これにより，要件④を満たすことが期待できる．

このアプローチで階層状態遷移図の記述品質を評価する測定量を定義しようとしたとき，以下が技術的課題となる．

課題 1：状態および状態遷移の階層性評価の決定

課題 2：状態と状態遷移の規模の決定

5. 階層性を評価する方法の提案

ここでは，階層性を評価するための 2 つの測定量の定義を提案し，その理論的根拠について述べる．

5.1 状態の階層性評価

状態の階層性評価に，状態階層複雑度 Cs を利用する．

5.1.1 準備

ここでは，状態階層複雑度 Cs を定義するうえで利用する要素とその定義を示す．

(1) 状態の階層レベル

状態の階層のネストの尺度となる階層レベルを定義する．ある状態 X の階層レベルを $L(X)$ で表す．階層レベルの定義は以下である．

- $X = U$ の場合： $L(U) = 0$
- $X = \text{OR 状態}$ の場合： $L(X) = L(S_P(X)) + 1$
- $X = \text{AND 状態}$ の場合： $L(X) = L(S_P(X))$

ここで， $S_P(X)$ は X の親状態， U はすべての状態からなる集まりであり，状態遷移図上ではいかなる状態によっても包含されていない領域を指す．本論文において階層状態遷移図の測定量を定義する際， U を一番外側の状態として扱うものとする．

階層レベルが異なる場合， $L(X) - L(Y) = n$ で， $n > 0$ の場合， X は Y より階層レベルが深いと呼ぶこととする．逆に $n < 0$ の場合は浅いと呼ぶ．

(2) 状態の重み

階層状態遷移図において，ある状態 X の状態の重み $w(X)$ を以下のように定義する．

- $X = U$ の場合： $w(U) = 1$
- $X = \text{OR 状態}$ の場合：
 $w(X) = w(S_P(X))/\text{noc}(S_P(X))$
- $X = \text{AND 状態}$ の場合： $w(X) = w(S_P(X))$

ここで $\text{noc}(X)$ は状態 X の直下の OR 状態数を表す．

状態の重みは，階層状態遷移図における各状態の重要度を表そうとしている．OR 状態に関する状態の重みは，各レベルの状態分割が，親状態の空間を OR 状態の個数で等分に分割していると考えられる．階層が深くなれば重みは減っていくようになる．

一方，AND 状態の重みは，親状態から継承されると考え，親状態と同じに設定している．図 7 に AND 状態を含む階層状態遷移図に対する重みの計算結果の例を示す．

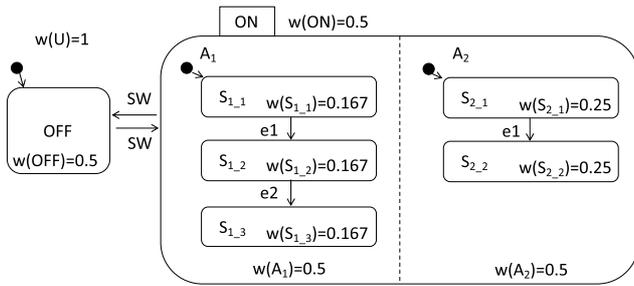


図 7 状態の重みの例

Fig. 7 Example of how to assign a weight to each state.

5.1.2 定義

ある階層状態遷移図の状態階層複雑度 C_s を、状態 X の階層複雑度 $C_s(X)$ を用いて再帰的に定義する。

- $C_s = C_s(U)$
- 状態 X について、子状態の集合 $\text{Sub}(X) = \{X_1, X_2, \dots, X_N\}$ とすると、 X の状態階層複雑度 $C_s(X)$ は：
 - ▶ $\text{Sub}(X) = \emptyset$ の場合： $C_s(X) = 0$
 - ▶ $N > 0$ かつ $\text{Sub}(X)$ が OR 状態の場合：

$$C_s(X) = (N - 1)w(X) + \sum_{i=1}^N C_s(X_i)$$

- ▶ $N > 0$ かつ $\text{Sub}(X)$ が AND 状態の場合：

$$C_s(X) = \sum_{i=1}^N C_s(X_i)$$

つまり、状態階層複雑度は、状態空間の分割 (OR 分割) の個数に、階層化による状態の重みを掛けて、足し合わせていくことで計算する。

5.1.3 設計根拠

状態階層複雑度 $C_s(X)$ は、階層的な状態の設計においてモジュール性を測定するための状態の複雑度として定義した。以下にその設計根拠を示す。

- ① 状態 X の複雑度は、 X の OR 分割数に比例すると考える。そのため、 X の直下への 2 つ目以降の OR 状態の追加で $C_s(X)$ は増加する。
- ② 状態 X への OR 状態の追加により、状態 X の複雑度は X の重み分だけ増加すると考える。階層レベルが深い状態での複雑度の増分は小さくなる。抽象状態の導入は注目する範囲を限定しモジュール性を高める効果があるため、この点を評価するための重み付けである。
- ③ OR 分割とは異なり、AND 分割は親状態のサブセット群を作らないため、状態 X の直下の AND 状態は状態 X の重みをそのまま引き継ぐものとする。このため、状態 X への AND 状態の追加によって状態 X の複雑度も重みも変化しない。

5.1.4 例による検証

(1) 等価な単純状態遷移図との比較による検証

図 1 と図 2 の 2 つの例を使って、状態階層複雑度が望ま

表 3 図 1 と図 2 に対する各複雑度測定量の値

Table 3 Complexity measurements for Figs. 1 and 2.

複雑度測定量	OR 状態 (図 1)		AND 状態 (図 2)	
	単純(a)	階層(b)	単純(a)	階層(b)
状態数	4	5	9	8
末端状態数	4	4	9	6
状態階層複雑度 C_s	3	2	8	4

しい指標となっているかを評価する。

表 3 は、図 1 と図 2 に対応する単純状態遷移図と階層状態遷移図について、状態数、末端状態数、状態階層複雑度をそれぞれ計算した結果である。

表 3 の 4 つの C_s の導出過程を以下に示す。

$$C_s(\text{図 1 (a)}) = (\text{noc}(U) - 1) \cdot w(U) = (4 - 1) \cdot 1 = 3$$

$$\begin{aligned} C_s(\text{図 1 (b)}) &= (\text{noc}(U) - 1) \cdot w(U) + (\text{noc}(\text{ON}) - 1) \cdot w(\text{ON}) \\ &= (2 - 1) \cdot 1 + (3 - 1) \cdot 0.5 = 2 \end{aligned}$$

$$C_s(\text{図 2 (a)}) = (\text{noc}(U) - 1) \cdot w(U) = (9 - 1) \cdot 1 = 8$$

$$\begin{aligned} C_s(\text{図 2 (b)}) &= C_s(\text{運転モード}) + C_s(\text{風速モード}) \\ &= (\text{noc}(\text{運転モード}) - 1) \cdot w(\text{運転モード}) \\ &\quad + (\text{noc}(\text{風速モード}) - 1) \cdot w(\text{風速モード}) \\ &= (3 - 1) \cdot 1 + (3 - 1) \cdot 1 = 4 \end{aligned}$$

図 1 (OR 状態の例) については、状態数では単純状態遷移図が優位、末端状態数は同等という評価になる。一方 C_s は、単純 (a) が 3 に対して、階層 (b) が 2 と小さくなり、逆に階層状態遷移図が優位となる。つまり、 C_s が抽象状態を設けてモジュール性を高めたという評価になっている。

図 2 (AND 状態の例) については、階層 (b) の方が全項目で値が改善している。AND 状態は元々、状態数の爆発を減らす目的で導入されているので、並列化の効果は、状態数と末端状態数でも十分に評価できることが分かる。

(2) 図 4 の例による検証

図 4 は、同じ仕様に対する記述品質が悪い例 (a) と良い例 (b) になっている。これらに対する状態階層複雑度 C_s の導出過程は以下のようなになる。

$$\begin{aligned} C_s(\text{図 4 (a)}) &= (\text{noc}(U) - 1) \cdot w(U) + C_s(\text{圧縮機作動中}) \\ &= (3 - 1) \cdot 1 \\ &\quad + (\text{noc}(\text{圧縮機作動中}) - 1) \cdot w(\text{圧縮機作動中}) \\ &= 2 + (2 - 1) \cdot 0.33 = 2.33 \end{aligned}$$

$$\begin{aligned} C_s(\text{図 4 (b)}) &= (\text{noc}(U) - 1) \cdot w(U) + C_s(\text{ON}) \\ &= (2 - 1) \cdot 1 + (\text{noc}(\text{ON}) - 1) \cdot w(\text{ON}) \\ &= 1 + (3 - 1) \cdot 0.5 = 2.0 \end{aligned}$$

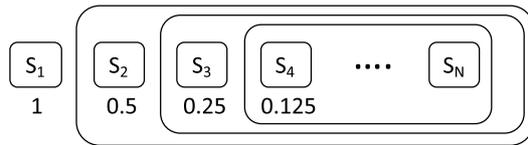


図 8 状態の 2 分割法

Fig. 8 Bisection method of states.

図 4 (a) の悪い例に対する C_s の値は $C_s = 2.33$ であり, (b) の良い例に対する C_s の値は $C_s = 2.0$ となる. この結果は, 状態階層複雑度 C_s が, 上位階層の状態数が少ないほど複雑度は低いと評価するためである.

5.1.5 値域に関する考察

状態階層複雑度 C_s の値域について考察する. N 個の状態を持つ単純状態遷移図を考える. このとき N は, 階層最上位レベルにおいて OR 状態が 2 つ以上あることを想定している. その場合, 状態階層複雑度 C_s の最小値は 1 になる. したがって, 状態階層複雑度 C_s の値域は以下となる.

$$1 \leq C_s < \text{単純状態に還元した場合の状態数}$$

つまり, 状態遷移複雑度の下限値は固定値 1, 上限値は仕様規模に依存する.

N 個の OR 状態を分割する 2 分割法は, 図 8 に示すように, 1 つの状態とそれ以外の状態とに分け, それ以外の状態を包含する抽象状態を再帰的に導入していく階層化の方法である. 単純状態遷移図は, この分割法で必ず階層状態遷移図を構成できるので, C_s の最小値は 2 より小さくなるのが分かる.

5.2 状態遷移の階層性評価

状態遷移の階層性評価として, 状態遷移複雑度を定義する.

5.2.1 定義

状態遷移複雑度 C_t を, 以下のように定義する.

$$C_t = \text{状態遷移数} + \text{状態またぎ数} \quad (3)$$

ここで, 状態遷移数は, OR 状態または AND 状態を遷移元とする状態遷移のみを対象としてカウントし, 初期状態やヒストリを遷移元とする状態遷移は含まないものとする.

5.2.2 設計根拠

状態遷移複雑度は, 階層的な状態の設計において実装容易性とモジュール性のバランスを測定するための遷移の複雑度である. 状態遷移複雑度は, 階層性を導入したことによって生じる以下の効果と弊害を測定するものである.

- ① 状態遷移数の削減 (実装容易性向上の効果)
- ② 状態遷移が抽象状態をまたぐ数の増加 (モジュール性の悪化の弊害)

OR 分割におけるモジュール性とは, 抽象状態が, その内外の境界を越えて構成要素間の変更影響が及ばないよう

表 4 図 3 と図 4 に対する各複雑度測定量の値

Table 4 Complexity measurements for Figs. 3 and 4.

複雑度測定量	単純(図 3)	階層(図 4)	
		悪い例(a)	良い例(b)
状態遷移数	13	12	5
状態またぎ数	0	5	0
状態遷移複雑度 C_t	13	17	5

に構成されている度合いである. これは, 各階層レベルの閉包性, すなわち階層レベル内で, イベント, 変数, 状態遷移が閉じていることにより達成される. モジュール性が高いほど各階層レベル間の依存関係は減るため, 各階層レベルでの変更の独立性が確保される. また段階的な仕様の確認を積み重ねることで仕様全体の確認がしやすくなる. 状態遷移が抽象状態をまたぐとこのモジュール性は損なわれる.

5.2.3 例による検証

表 4 は, 図 3 と図 4 について, 状態遷移数, 状態またぎ数, 状態遷移複雑度 C_t を測定した結果である.

階層 (a) の状態遷移複雑度 C_t は 17 となっており, 単純状態遷移の 13 に対して, 4 ポイント増加している. これは, 状態遷移数の減少分 1 を, 状態またぎ数の増加分 5 が超えていることにより, 複雑度が増したと評価された結果である.

これに対して, 階層 (b) の状態遷移複雑度 C_t は 5 となっており, 単純状態遷移の 13 に対して, 8 ポイント減少している. これは, 階層 (b) が状態またぎ数 0 のため, 状態遷移数の減少分 8 がそのまま複雑度の減少として評価されたためである.

このように, 状態遷移数は, 階層性をうまく使っていると統合されるので減少し, 状態またぎ数は, うまく使えていないと増加する. つまり, 状態遷移複雑度 C_t は, 抽象状態の導入による実装容易性の向上とモジュール性の悪化のバランスをみる測定量となっている.

状態階層複雑度 C_s は, 状態の 2 分割法によりその仕様における最小値に持っていくことは可能であるが, その場合, 反対に階層状態遷移複雑度 C_t が増加してしまう. 状態遷移複雑度を導入することで, そうした抽象状態導入の負の面を評価することができる.

5.2.4 値域に関する考察

状態遷移複雑度の値域は, 抽象状態を除いた等価な単純状態遷移図の状態遷移数を基準にして, 抽象状態を入れたことにより状態またぎが生じる数 (プラス) と, 遷移元または遷移先が同じで, イベントおよびガード条件を共通化できる状態遷移の総数 (マイナス) のある範囲になる. 元々状態遷移数の上限は仕様規模により決定し, 1 つしか状態を含まない抽象状態を許さないとすれば, 導入可能な抽象

状態数は限られるので状態またぎ数の上限も仕様規模により決定する。つまり、状態遷移複雑度の上下限値は、仕様規模に依存する。

5.3 正規化のための規模指標

5.3.1 提案する規模指標

これまでに、「状態の階層性評価」のための状態階層複雑度 C_s と、「状態遷移の階層性評価」のための状態遷移複雑度 C_t を定義し、その理論的な根拠を示してきた。しかし、 C_s は上限値、 C_t は上下限値ともに、仕様規模とともに増大する。そのため、これらの指標だけでは、単一の階層状態遷移図の記述品質の良し悪しを評価できない。

この問題を解決するため、状態階層複雑度と状態遷移複雑度を正規化するために使うことができる規模指標として以下の2つを提案する。

状態規模 N_s : 階層性がなかった場合の OR 状態の数

状態遷移規模 N_t : 階層性がなかった場合の状態遷移数

状態規模 N_s および状態遷移規模である N_t は、階層性がなかった場合の状態遷移図を想定し、それぞれ、総状態数と総状態遷移数となる。

5.3.2 階層性がなかった場合の状態遷移図の構成法

「階層性がなかった場合の状態遷移図」は、OR 状態と AND 状態をどのように展開するかという問題に帰着する。以下、AND 展開と OR 展開の方法について述べる。AND 展開と OR 展開は、階層状態遷移図における抽象状態がなくなり、AND 状態が最上位にくるまで繰り返し適用する。

(1) OR 状態の展開

抽象状態を除去する手順を以下に示す。

- ① 出ていく状態遷移をすべての直下状態へ展開する。
- ② 入る状態遷移は直下の指定された状態へ展開する。
- ③ 抽象状態を除去する。

図 9 の右の状態遷移図は、左の階層状態遷移図から抽象状態「ON」を削除した結果となっている。その際、ON から OFF へ向かっている状態遷移は、冷房、暖房、送風それぞれへ複製され展開され、ON から冷房、暖房、送風へ遷移している状態遷移は、それぞれの下の状態へ展開されている（手順①）。また、OFF から出ている状態遷移は初期状態へ遷移するので、冷房状態へと展開される（手順②）。そのうえで状態「ON」を削除する（手順③）。

図 9 右図が、「階層性がなかった場合の状態遷移図」となり、その状態数が N_s 、状態遷移数が N_t となる。規模指標値は、 $N_s = 4$ 、 $N_t = 13$ となる。

(2) AND 状態の展開

AND 状態への展開は、すべての AND 状態を OR 階層の最上位まで展開する方法をとる。AND 状態を展開する手順は以下ようになる。

- ① AND 状態の外にある OR 状態を AND 状態のそれぞれに複製展開し、削除する。

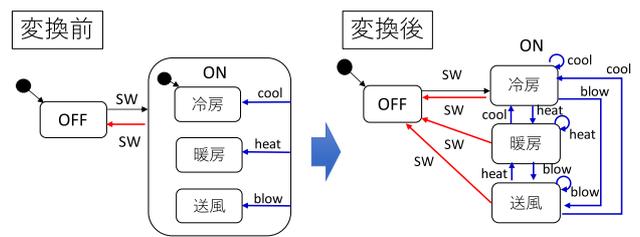


図 9 OR 状態の展開方法

Fig. 9 Method for expanding OR-states.

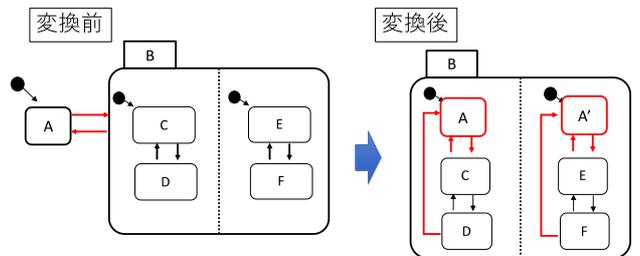


図 10 AND 状態の展開の例

Fig. 10 Example of expanding AND-states.

- ② AND 状態へ遷移する状態遷移は、その下に指定された状態（初期状態）へ展開する。
- ③ AND 状態から遷移する状態遷移は、その下のすべての OR 状態から複製先への遷移へ展開する。

図 10 の例では、① AND 状態の外にある状態 A を状態 B の中の 2 つの AND 状態内へ A および A' として複製し、元の A を削除する。次に② A から B へは、C と E はそれぞれ初期状態であるため、A から C と A' から E へは状態遷移に展開する。最後に③ B から A へは、C と D から A、E と F から A' へは状態遷移へ展開する。

AND 状態の階層状態遷移図から単純状態遷移図への変換も OR 状態の展開と同様に、最上位へ状態遷移しているため状態遷移数が増える形となる。さらに、OR 状態を各 AND 状態へ複製・展開しているため、状態数も増加する形となる。図 10 の例では、 $N_s = 6$ 、 $N_t = 10$ となる。

AND 分割の展開方法として図 2 のように AND 状態に含まれる状態の積を意味する状態に置換する方法が一般的であるが、階層状態遷移図の階層性を評価する測定量において、並列性の導入による測定量への影響を抑制するため本展開法を採用した。AND 分割を最上位の階層に集約する本展開法は、オブジェクト指向による世界の分割（オブジェクト分割）に準じたものとなっている。

なお、上記の AND 状態の展開は、状態と遷移の規模を考えるための展開であり、等価な状態遷移ではないことに注意が必要である。たとえば、上記の例で、変換後は AND 状態の左が A のとき、右は A' である必要があり、完全に独立でないこと、他にヒストリは一意に変換できないこと、など意味的な一致を達成できないことがある。

表 5 図 1 に対する階層性の品質指標の値

Table 5 Quality index values for hierarchical related to Fig. 1.

階層性の 品質指標	対象階層状態遷移図 (図 1)	
	(a)	(b)
Es	0.5 (= (3-1)/4)	0.25 (= (2-1)/4)
Et	1.0 (= 7/7)	0.71 (= 5/7)

5.4 提案する指標の定義

これまで定義してきた、状態階層複雑度 C_s 、状態遷移複雑度 C_t 、状態規模 N_s 、状態遷移規模 N_t に基づき、階層性の品質評価指標 E_s と E_t を以下のように定義する。

$$E_s = (C_s - 1) / N_s \tag{4}$$

$$E_t = C_t / N_t \tag{5}$$

ここで、 E_s において 1 を引いているのは、 C_s の最小値 1 を引き、 E_s の最小値を 0 に補正するためである。

表 5 は、図 1 に対する階層性の品質指標の値を計算した結果である。

6. 実験による評価

6.1 実験目的

提案する品質評価法の要件②「単独の仕様に対して記述品質を評価できる」の達成を確認するためには、2つの測定量が、問題依存しない品質基準を設定できるかを調査する必要がある。そこで、以下の調査点を設定し実験により評価した。

- RQ1. 異なる問題の記述群が偏りのない分布を持つか。
- RQ2. 過去のデータに基づき品質基準値を設定できるか。

6.2 実験方法

上記の実験目的のため、複数の問題とそれらに対する複数の解答群(仕様記述群)を準備した。表 6 に実験に用いた問題、記述実験の被験者、問題規模、標本数を記す。

エアコンと自動車の記述実験の被験者は、異なる年度の授業の中で、階層状態遷移図の文法と記述例などの紹介を行い、小問題の解答を実施した状態で、記述を実施した。問題に対する誤解を極力減らすために、対応するテストケースを配布し、授業の中でレビューを行い、修正を入れている。それでも、仕様を満たしていないものや文法上の間違いのあるものは修正を施したうえでサンプルとして採用した。この際、修正により状態の階層化構造に変更が出てしまうものは除外した。

6.3 実験結果

「RQ1. 異なる問題の記述群が偏りのない分布を持つか」を調査するために、 E_s と E_t を散布図としてプロットしたのが図 11 である。

表 6 実験に用いた問題と被験者と標本数

Table 6 Data about statecharts which each subject wrote in the experiment.

問題	被験者	問題規模(平均)		標本数
		状態数	遷移数	
エアコン	設計仕様技術講座の受講生, 教員	22.2	31.0	5
自動車	設計仕様技術講座の受講生, 教員	40.1	62.7	7
オンラインショッピング	教員, 学生(出題者)	33.0	60.5	2

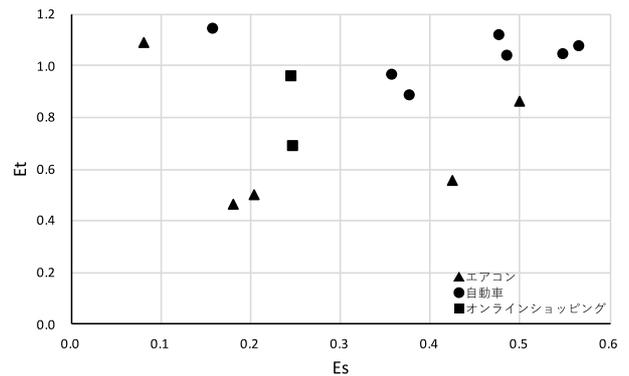


図 11 提案した指標による散布図

Fig. 11 Scatter chart for specimen.

この結果、以下の結果を得た。

- 同一グラフ(散布図)にすべてのデータがスケールの歪みなくプロットできている。
- 標本数が多いエアコンと自動車について、それぞれ E_s が 0.08~0.50, 0.16~0.57, E_t が 0.46~1.09, 0.88~1.14 と、重なって分布している。
- 仕様規模の大きい自動車が、小さいエアコンの方に比べ、 E_t のばらつきの範囲が狭い傾向がある。これに対して、自動車と同程度の仕様規模のオンラインショッピングは E_t が 0.69~0.96 と範囲がより広がっている。このことから、このばらつきの傾向は、仕様の規模の影響より生じたものではなく、たとえば、自動車が、独立性の高いオブジェクト分割、つまり並列性を選択しやすい問題であったことなど、問題の性質によるものと推測できる。

以上の結果から、正規化により、仕様規模に依存せず、提案測定量により、2次元平面上の位置で階層状態遷移図の階層性の品質を評価できる見込みがあることが分かった。これにより、設計者が記述する階層状態遷移図の階層性の品質を、比較対象なく評価することが可能になる。

6.4 基準値の設定

「RQ2. 過去のデータに基づき品質基準値を設定できるか」を調査するために、状態遷移と状態に関する階層性の

品質指標の基準値設定について検討した。

(1) 状態遷移に関する階層性の品質評価指標

状態遷移に関する階層性の品質評価指標 Et については、以下の基準値を用いるのがよいと考える。

$$Et \leq 1.0$$

その理論的根拠は、 Et が、同じ問題を記述する「階層性のない状態遷移図」の状態遷移数を分母に持つので、基準値 1.0 は階層性を持ち込んだことによる状態遷移数の削減数と状態またぎ数が相殺された状態と考えられる。1.0 以上の場合、階層性のない状態遷移図の方がむしろ良好ということになる。

(2) 状態に関する階層性の品質評価指標

状態に関する階層性の品質評価指標 Es については、以下の基準値を用いるのがよいと考える。

$$Es \leq 0.4$$

この基準値に関しては、 Et に関するものと異なり理論的な根拠はなく、今回の記述実験のサンプルの特徴を見ただけで設定した仮値である。具体的には、 Es が 0.4 以上のサンプルについて、0.4 未満のサンプルには見られない、「最上位レベルに AND 状態が横並びで存在し、かつほとんど階層を用いていない」という特徴が見られたことを根拠としている。この基準の適用により、AND 分割の安易な使用を諫め、ON/OFF のような共通性を抜き出す OR 分割の適用の可能性を考えることを促すなどの効果があると考えられる。サンプル数が増えれば、こうした階層性の良否を判定する適切な基準値を設定できると期待できる。

図 12 は、散布図を 2 つの基準値により 4 つの領域に分割し、基準値の内側に存在するサンプル ($Et \leq 1.0$ かつ $Es \leq 0.4$) は、記述品質の良い状態遷移図であることを示しており、それ以外は、状態の階層性が適切でないもの、状態遷移の階層性が適切でないもの、両方が適切でないものに分類できるようになった。

6.5 提案する指標に関する考察

変換後の階層性がない場合の状態数 (Ns) および状態遷移数 (Nt) は、5.3.2 項の変換によって生じる等価な階層状態遷移図の中では、それぞれ Cs , Ct 以上となる。提案手法は、これを正規化のための分母としている。 Es と Et とでは、正規化により評価できることが異なる。

- $Es = (Cs - 1) / Ns$: Cs と Ns は同じ仕様を基にしているため、この正規化により Es は規模に依存しない 0 から 1 の範囲の数値となる。 Es に適切な基準値を設定できれば、基準値に対する大小により、対象状態遷移図の状態の階層性の良否を単独で判断できる。
- $Et = Ct / Nt$: Ct が対象の「状態遷移数+状態またぎ数」なので、正規化により、状態遷移数の減少分と状

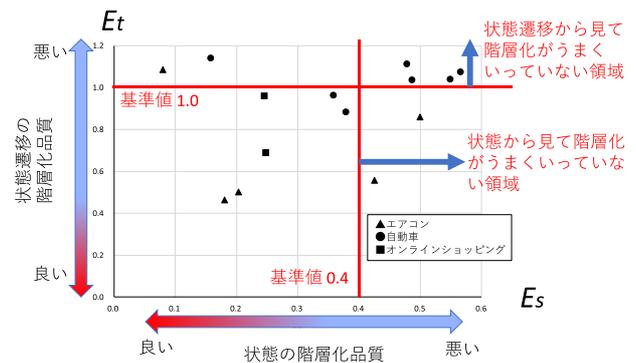


図 12 基準値を設定した散布図

Fig. 12 Scatter chart with two reference values.

態またぎ数の増加分のどちらが多いかを基準値 1.0 に対する大小で判断できる。この正規化により、 Et と基準値 1.0 の間の差分が規模に依存しなくなるので、この差分値により対象階層状態遷移図の状態遷移の階層性の良否だけでなく、良否の度合いも単独で判断できる。

上記のように、提案手法は、階層状態遷移図の階層性の良否を、 Es , Et がそれぞれに対して設定された基準値に対して評価できる。

7. 関連研究

階層状態遷移図の複雑さには、種々の要因がある。Beldjehem [6] は、階層状態遷移図の複雑さの 3 つの分類：計算的複雑さ、心理的複雑さ、表現的複雑さを提供し、心理的複雑さをさらに問題固有の複雑さ、設計者の認知能力からくる複雑さ、構造的複雑さに分類している。これらは、分離して考える必要があることを示している。提案手法は、規模正規化を導入することで、構造的複雑さから問題固有の複雑さの影響を分離可能とする測定量を定義している。

階層状態遷移図の可読性を、記述要素の基本測定量と関連付ける研究もいくつか行われている。Miranda ら [7] は、階層状態遷移図のアクション数、状態数、状態遷移数と理解にかかる時間と相関があることを実験により示している。Cruz-Lemus ら [8] は、階層状態遷移図における複合状態の理解容易性への影響を大規模な実験によって評価し、複合状態の利用が読みやすさを改善することを示している。Fonte ら [9] は、状態遷移数、状態遷移数/状態、状態の深さの 1 次式で階層状態遷移図の複雑度の予測式を立て、開発者による複雑度の 5 段階の評価結果を用いて推定する実験を行っている。これらの研究は、問題固有の難しさを分離して評価することはしていない。

階層状態遷移図の記述品質の改善手法として、Abadi ら [10] が、グループ化、併合、抽出、引き上げ、複合の 5 種類の手法のリファクタリングパターンを提案している。記述品質が悪いと診断された記述を改善するために利用可

能である。

8. 結論と将来の課題

本論文では、単一の階層状態遷移図だけから、その記述品質を評価する手法を提案した。この評価手法は、状態と状態遷移の2つの視点から、視点別の複雑度測定量と規模測定量を定義し、それらを用いて規模正規化された記述品質の測定量を導出していることに特長がある。

記述実験より、提案する記述品質の評価手法が、異なる分野と規模の問題に対して測定値の分布の偏りが比較的小さいこと、および単一の階層状態遷移図を評価するために必要な基準値を設定できる見通しがあることが分かった。

要件④「測定値に基づき仕様に対する改善点が分析できる」に関しては、測定値の設計上の工夫を施したにすぎず実開発における評価はまだできていない。

今後の課題として、状態遷移図実験に用いた標本数が少ないため、指標の有効性の評価や適切な基準値の設定には、同じ問題の異なる記述や、異なる問題に対する記述に関して、標本数を積み上げることが必要である。また、本評価手法を多くの実開発へ適用し、階層状態遷移図の記述品質の改善に有用であることも示していく必要がある。

謝辞 実験とデータ処理を手伝っていただいた芝浦工業大学の中島研究室の鈴木淳志氏に、謹んで感謝の意を表す。

参考文献

- [1] 2009年度版組込みソフトウェア産業実態調査報告書, 入手先 (http://warp.da.ndl.go.jp/info:ndljp/pid/11241027/www.meti.go.jp/policy/mono.info.service/joho/downloadfiles/2010software_research/10project_houkokusyo.pdf) (参照 2019-10-05).
- [2] Harel, D.: Statecharts: A visual formalism for complex systems, *Science of Computer Programming*, Vol.8, No.3, pp.231-274 (1987).
- [3] Genero, M., Miranda, D. and Piattini, M.: Defining and validating metrics for UML statechart diagrams, *Proc. QA00SE 2002* (2002).
- [4] McCabe, T.J.: A Complexity Measure, *IEEE Trans. Softw. Eng.*, Vol.SE-2, No.4, pp.308-320 (1976).
- [5] Hall, M.: Complexity Metrics for Hierarchical State Machines, *SSBSE 2011*, LNCS-6956, pp.76-81 (2011).
- [6] Beldjehem, M.: A granular hierarchical multiview metrics suite for statecharts quality, *Advances in Software Engineering*, No.5 (2013).
- [7] Miranda, D., Genero, M. and Piattini, M.: Empirical validation of metrics for UML statechart diagrams, *Enterprise Information Systems V*, pp.101-108, Springer, Dordrecht (2004).
- [8] Cruz-Lemus, J.A., Genero, M., Manso, M.E. and Piattini, M.: Evaluating the effect of composite states on the understandability of UML statechart diagrams, *International Conference on Model Driven Engineering Languages and Systems*, pp.113-125, Springer, Berlin, Heidelberg (2005).
- [9] Fonte, D. et al.: Modeling Languages: Metrics and as-

sessing tools, *CoRR*, abs/1206.4477 (2012).

- [10] Abadi, M. and Feldman, Y.A.: Refactoring of Statecharts, LNCS 5831, pp.50-62 (2009).



松井 聡一

三菱電機株式会社。1990年名古屋大学大学院工学研究科電気工学・電気工学第二及び電子工学専攻博士前期課程修了。同年三菱電機(株)に入社。組込みソフトウェアの生産技術に関する研究・開発に従事。現在、三菱電機(株)人材開発センターにおいて人材育成の企画、社員の教育を実施。



徳本 修一 (学生会員)

三菱電機株式会社。1995年北海道大学大学院工学研究科精密工学専攻修士課程修了。同年三菱電機(株)入社。現在、同社情報技術総合研究所に勤務。ソフトウェア生産技術、プロジェクトマネジメントに関する研究・開発に従事。2019年大阪大学大学院情報科学研究科博士課程後期課程在学中。電子情報通信学会会員。



中島 毅 (正会員)

芝浦工業大学。1984年早稲田大学大学院修士課程修了。同年三菱電機(株)入社。2008年早稲田大学大学院博士課程修了, 博士(工学)。現在、芝浦工業大学情報工学科教授, ソフトウェア工学に関する研究に従事。著書に『IT Text ソフトウェア開発改訂第2版』(共著, オーム社)。技術士(情報工学/総合技術監理)。IEEE CS, 電子情報通信学会, 電気学会各会員。