

# 相互作用オカレンスに着目した シーケンス図の三次元化に関する研究

前澤悠太<sup>†</sup> 高田真吾<sup>†</sup>

<sup>†</sup>慶應義塾大学理工学部

## 1 はじめに

ソフトウェア開発や保守において、開発者はソフトウェアの振る舞いを表現するためにシーケンス図を用いることが多い。シーケンス図は、たとえ小規模なソフトウェアでも通常複数記述する必要があり、これらはソフトウェアの振る舞い毎に切り分けて記述される。そのため、開発者はあるシーケンス図に注目する時、それと関係のあるシーケンス図がどれで、どういった関係があるのか把握しにくい。そこで我々は、三次元空間上に複数のシーケンス図を配置して、それらの関係を描写するツールを提案する。これにより開発者は、あるシーケンス図に注目すると同時に、シーケンス図間の関係を視覚的に認識することが可能となる。

## 2 背景

オブジェクト指向のソフトウェア設計で一般的に用いられる表現記法として UML (Unified Modeling Language) がある。UML には、例えばクラス図やシーケンス図、状態遷移図といった複数の図が含まれている。

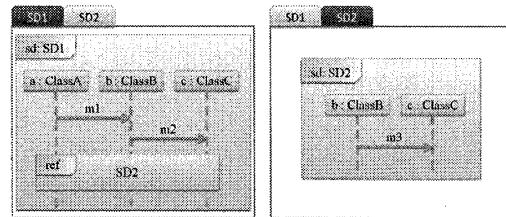
開発者はソフトウェアの振る舞いを表現するためにシーケンス図を用いることが多い。UML1.x から UML2.x への変更に伴い、シーケンス図は相互作用オカレンスや複合フラグメントを記述できるようになった。これらによりシーケンス図は、より膨大なシステムの振る舞いをより詳細に表現できるようになった。

### 2.1 二次元ツールでのソフトウェア開発

シーケンス図を取り扱うツールの例として Rational Software Architect<sup>1</sup>, Together<sup>2</sup>, astah<sup>\*3</sup>などが挙げられる。これらツールは一般的に図 1 に示すように、1 つの注目するシーケンス図を画面上に表示して操作する。タブバーにシーケンス図名が列挙されており、タブを選択することで表示するシーケンス図を切り替える。

### 2.2 二次元ツールでの一般的な問題点

開発者はあるシーケンス図に注目して変更を加える際、そのシーケンス図が参照するシーケンス図にどのように影響を与えるか把握しなければならない。また、



(a)シーケンス図SD1を表示している場合 (b)シーケンス図SD2を表示している場合  
図 1: 二次元ツールでシーケンス図を取り扱う例

他のどのシーケンス図が変更対象のシーケンス図を参照している、どのように影響が及ぶのかもまた考慮しなければならない。図 1 のシーケンス図 SD1 と SD2 を例に挙げる。SD1 に注目して操作する場合、SD1 が参照している SD2 が表す振る舞いを把握していなければならない (図 1(a))。また、SD2 に注目して操作する場合、SD2 は SD1 に参照されており、SD1 が表す振る舞いを把握していなければならない (図 1(b))。つまり、開発者は参照元・参照先のシーケンス図とその参照関係を把握していなければならない。しかし二次元上で表現されるシーケンス図では、複数のシーケンス図とそれらシーケンス図間の参照関係を表示することはできない。

## 3 UML 図の三次元化に関する関連研究

UML の開発者である Booch らにより、UML は三次元表現可能である、と言われている [1]。そこで、Gogolla らはメッセージのやり取りに着目してシーケンス図の三次元化を行った [2]。UML1.x のシーケンス図を二次元上で記述する場合、複数のオブジェクトへ同時にメッセージを送信する記述を UML 規則に則って記述できないという問題がある。Gogolla らはこれを複数のオブジェクトを三次元空間上に配置することにより解決した。しかし、UML2.x のシーケンス図は、複数の流れの同時実行を表す複合フラグメント par を記述することができる。Gogolla らの手法は UML2.x のシーケンス図に対応しておらず、また複数のシーケンス図を記述することを考慮していないという問題がある。また、McIntosh らはコンポジット状態に着目して状態遷移図の三次元化に取り組んでいる [3]。状態遷移図は、コンポジット状態という状態遷移を入れ子にした状態を記述することができ、階層構造的に取り扱うことができる。構造を構成する要素に着目して図の三次元化を行うことは有用であると考えられる。

### 3D Visualization of Sequence Diagrams focusing on Interaction Occurrence

Yuta MAEZAWA<sup>†</sup> and Shingo TAKADA<sup>†</sup>

<sup>†</sup>Faculty of Science and Technology, Keio University

<sup>1</sup><http://www-06.ibm.com/software/jp/rational/products/rsd/>

<sup>2</sup><http://www.borland.com/jp/products/together/>

<sup>3</sup><http://astah.change-vision.com/ja/>

#### 4 提案：三次元シーケンス図エディタ

UML2.x のシーケンス図は、相互作用オカレンスをを用いてシーケンス図間の参照関係を記述できる。相互作用オカレンスを用いることにより、複数のシーケンス図を構造化して管理することができる。第 2.2 節で述べた通り、二次元ツールでは複数のシーケンス図とそれらシーケンス図間の参照関係を表示できないという問題がある。そこで、複数のシーケンス図を三次元空間上に配置して、相互作用オカレンスによるシーケンス図間の参照関係を表示するツールを提案する。これにより、開発者はあるシーケンス図に注目すると同時に、シーケンス図間の関係を把握することができる。

##### 4.1 相互作用オカレンスの表示

シーケンス図は、左上に sd と示されたフレーム内にオブジェクトとそれらの振る舞いを記述する。相互作用オカレンスは、左上に ref と記述されたフレームであり、中央部には参照先のシーケンス図名が記述される。相互作用オカレンスのフレームは、参照先のシーケンス図のフレームを最小化したものであると考えられる。そこで、相互作用オカレンスのフレームと参照先のフレームとのリンクを表示する。

図 2 に図 1 のシーケンス図を三次元化した例を示す。図 2 では、手前にシーケンス図 SD1 が配置されており、奥にシーケンス図 SD2 が配置されている。SD1 に相互作用オカレンスが記述されており、この相互作用オカレンスは SD2 を参照している。本ツールは、図 2 で示すように相互作用オカレンスのフレームと参照先のシーケンス図のフレームとのリンクを表示する。

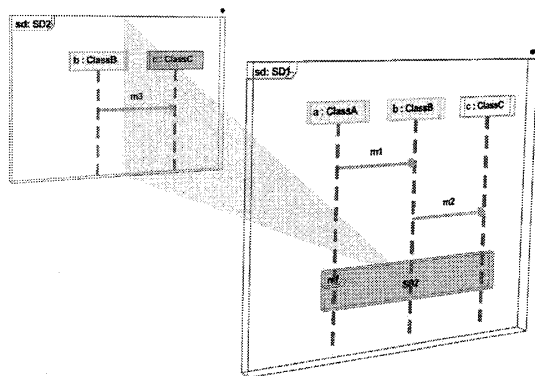


図 2: 相互作用オカレンスの表示の例

例えば、開発者がシーケンス図 SD1 に注目して操作する場合、メッセージ m1, m2 が送信された後に、注目するシーケンス図をシーケンス図 SD2 に切り替えなくても、相互作用オカレンスの参照先の SD2 でメッセージ m3 が送信されることがわかる。開発者が SD2 に注目して操作する場合、注目するシーケンス図を他のシーケンス図に切り替えなくても、SD2 は SD1 に参照されており、メッセージ m3 が送信される前にメッセージ m1, m2 が送信されることがわかる。

##### 4.2 三次元化による問題の解決

複数のシーケンス図を三次元空間上に配置することで、いくつか新たな問題が生じる。

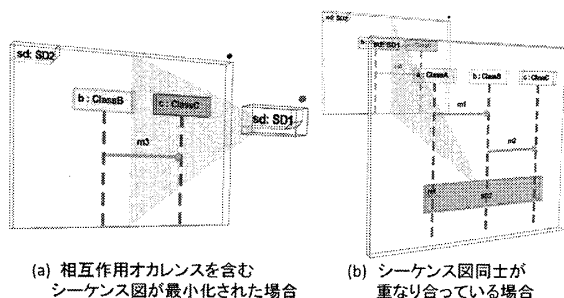


図 3: 三次元化の問題の解決の例

##### 4.2.1 シーケンス図の最小化

1 つの画面に複数のシーケンス図を表示させるため、表示される情報が過多になる可能性がある。これを回避するために、シーケンス図を最小化させて情報をフィルタリングする機能を実装した。図 3(a) にシーケンス図 SD1 が最小化された場合の例を示す。最小化されたシーケンス図は、シーケンス図のフレームのみが表示され、ライフラインやメッセージ、複合フラグメントなどはフィルタリングされる。相互作用オカレンスもフィルタリングされるため、相互作用オカレンスのリンク元となるフレームは最小化されたシーケンス図のフレームに切り替えられる。つまり、参照元のシーケンス図がどのシーケンス図かという情報のみが見える。

##### 4.2.2 シーケンス図の移動

図 3(b) のようにシーケンス図同士が重なり合い、シーケンス図に記述された情報が読み取りにくくなる可能性がある。これを回避するために、シーケンス図の配置位置を移動させる機能を実装した。

#### 5 結論

相互作用オカレンスに着目してシーケンス図の三次元化を行った。これにより、開発者は構造化した複数のシーケンス図間を見通すことができるようになり、システムの振る舞いに対する開発者の理解を助けることができる。

課題としては、より効果的なレイアウトやナビゲーションを追及する必要がある。また、実際にシステムの開発プロジェクトに導入して開発者の評価を得る必要がある。

##### 参考文献

- [1] Booch, G., Rumbaugh, J., Jacobson, I.: "UML Summary (Version 1.3)", Rational Corporation, Santa Clara, pp. 489-497, 1999.
- [2] Gogolla, M., Radfelder, O., Richters, M.: "Towards Three-Dimensional Animation of UML Diagrams", Proceedings of the 2nd International Conference on UML, pp. 489-502, 1999.
- [3] McIntosh, P., Hamilton, M., Schyndel, R.: "X3D-UML: 3D UML State Machine Diagram", Proceedings of the 11th International Conference on MODELS, pp. 264-279, 2008.