

# メソッド呼び出しパターン分析ツールの提案と実現

A Tool for Analyzing Method Call Patterns in Source Code

山田 吾郎\* 吉田 則裕† 三宅 達也‡ 宮崎 宏海§ 井上 克郎¶

あらまし メソッド呼び出しパターンとは、ソースコードの複数箇所に出現する類似した構造をもつコード記述のことである。本稿では、メソッド呼び出しパターンの分析を支援するツールの提案と実現について述べる。

**Summary.** Method call pattern is a sequence of method calls which appears plural times in source code. In this paper, we propose a tool that supports analysis of method call patterns.

## 1 背景

**メソッド呼び出しパターン** (以下パターン) とは、ソースコードの複数箇所に出現する類似した構造をもつコード記述のことである [1]。例えば、ファイルの読み込み処理を考える。まずファイルを開くためのメソッドを呼び出し、次にファイルからデータを読み込み、最後にファイルを閉じるメソッドを呼び出す。このようなファイル読み込み処理が複数箇所に出現したとき、“ファイルを開く”メソッドと“ファイルを閉じる”メソッドがパターンである。また、ソースコード中に現れるパターンの実体をパターンのインスタンス (以下インスタンス) という。パターンは、ソースコードの類似性を示す性質をもち、また、特定のメソッドの利用例とも捉えることができる。

我々の研究グループはこのようなパターンの検出ツールとして **Fung** [2] を開発した。検出結果にはいくつかの用途が考えられており、そのうちの1つにリファクタリング [3] 対象の発見が挙げられる。複数のソースコードに分布するインスタンスは、同様の処理をいくつかのクラスが行っている事を示す場合があり、共通の親クラスが存在すればそのクラスに引き上げるなど、リファクタリングの対象となり得る。また、頻出する実装方法の理解支援も挙げられる。例えば、ファイル操作について理解する場合、ファイルを開くメソッドのみの理解では不十分であり、ファイル操作やファイルを閉じるためのメソッドも知る必要がある。ファイル操作に関連するパターンにはファイル操作に必要なメソッドが含まれている可能性が高く、理解支援に役立つ。

しかし、Fung の機能はパターンの検出のみであり、検出結果をこれらの用途に利用するためには手作業による分析しか方法がない。検出対象のソースコードが大規模になるに従い、パターンの数が増大する可能性が高く、分析がより困難になると考えられる。このため、検出結果を分析するツールが必要である。

## 2 メソッド呼び出しパターン分析ツール

我々は新たにメソッド呼び出しパターン分析ツール (以下本ツール) を開発した。図1に本ツールのスナップショットを示す。本ツールは、スキータプロット、パターンリスト、ソースコードビューから構成されている。スキータプロットは、左上を原

\*Goro YAMADA, 大阪大学基礎工学部情報科学科

†Norihito YOSHIDA, 大阪大学大学院情報科学研究科

‡Tatsuya MIYAKE, 大阪大学大学院情報科学研究科

§Hiromi MIYAZAKI, 大阪大学大学院情報科学研究科

¶Katsuro INOUE, 大阪大学大学院情報科学研究科

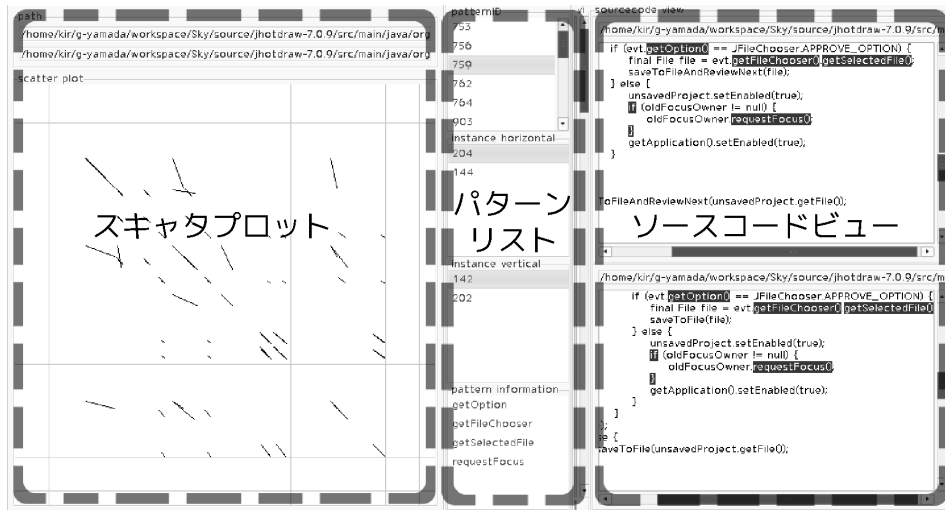


図1 本ツールのスナップショット

点とし、縦軸と横軸にソースコードをファイルパス順に並べ、同一のパターンに属するインスタンス同士の交点に点を描写した図である。スカタプロットを閲覧することで、ソースコードのどの部分とどの部分が同じパターンのインスタンスであるかを俯瞰的に知ることができる。スカタプロット中の任意の点を選択することで、対応する縦軸と横軸のソースコードをソースコードビューに表示し、それらソースコードに含まれるパターンについての情報をパターンリストに表示することができる。パターンリストは、スカタプロットで選択された2つのソースコード中に共通して存在するパターンID、および縦軸と横軸それぞれのインスタンスを列挙し、さらにパターンに含まれるメソッドのリストを表示する。パターンIDを選択することで、ソースコードビューが表示するファイル内に存在する全てのインスタンスを強調表示(ハイライト)する。

また、スナップショットには含まれていないが、パターンのフィルタリング機能を実装している。この機能は任意のパターンとそのインスタンスについて、スカタプロットとパターンリストに表示するか否かを手動で変更することができる機能である。

### 3 今後の課題

特定のメソッド呼び出しを含むパターンを検索する機能の実装を計画している。この機能を実現することで、あるメソッドとともに呼び出されるメソッドを知りたい開発者を支援することができる。

また、パターンのインスタンス数や、パターンに含まれるメソッドの数でフィルタリングする機能を実現し、開発者が必要とするパターンのみを選択可能にする予定である。

### 参考文献

- [1] 三宅 達也, 石尾 隆, 谷口 考治, 井上 克郎: メソッド呼び出しパターンとして現れる横断的関心事の検出, 電子情報通信学会技術研究報告, Vol.107, No.99, SS2007-9, pp. 1-6, 2007.
- [2] Fung: A pattern mining tool for java method calls, <http://sel.ist.osaka-u.ac.jp/~ishio/fung/>.
- [3] M. Fowler: Refactoring: improving the design of existing code, Addison Wesley, 1999.