

Refactoring Graphs: Assessing Refactoring over Time

(リファクタリンググラフ：時間の経過に伴う
リファクタリングの評価)

Aline Brito, Andre Hora, Marco Tulio Valente

2020 IEEE 27th International Conference on
Software Analysis, Evolution and
Reengineering (SANER)

リファクタリンググラフ

- 経時的なリファクタリング活動を表示
- 連結でない部分グラフの集合
 - $G' = (V', E')$
- 各 G' はリファクタリング部分グラフと呼ばれ, ノードの集合 V' と有向エッジの集合 E'
- メソッドを頂点, 行われたリファクタリング操作を辺とした

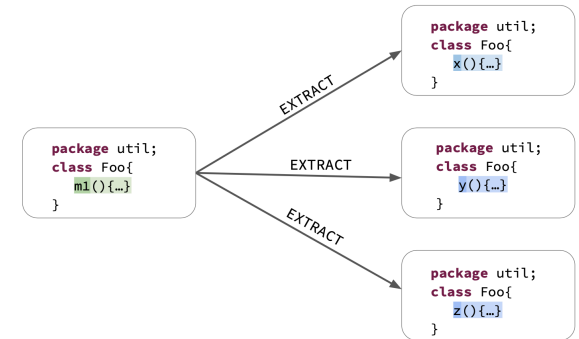


Fig. 1. Refactoring subgraph produced by only one developer

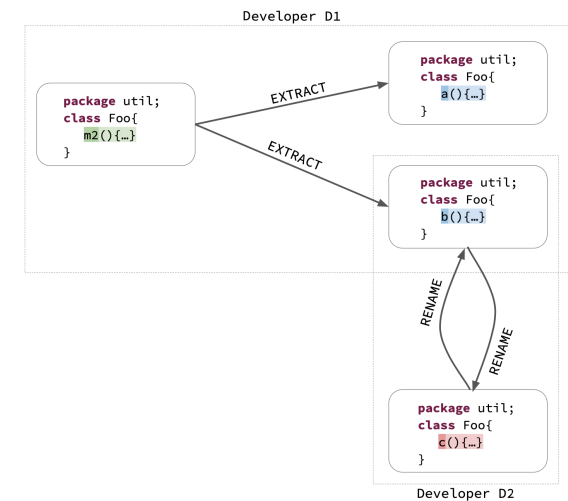


Fig. 2. Refactoring subgraph over time

リファクタリンググラフ

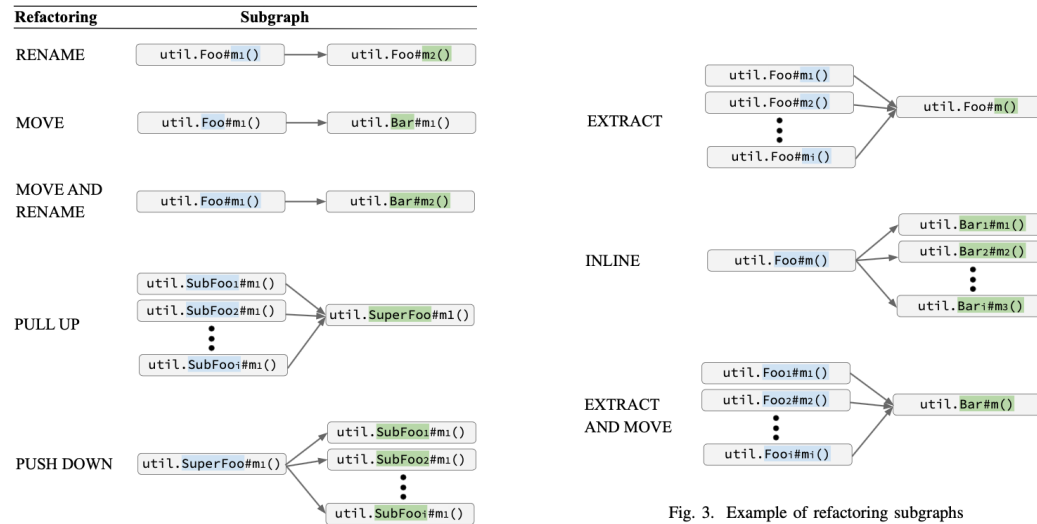
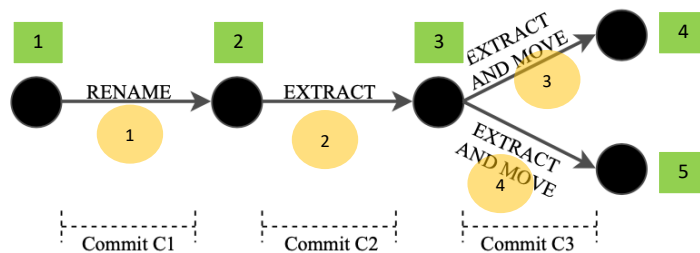


Fig. 3. Example of refactoring subgraphs

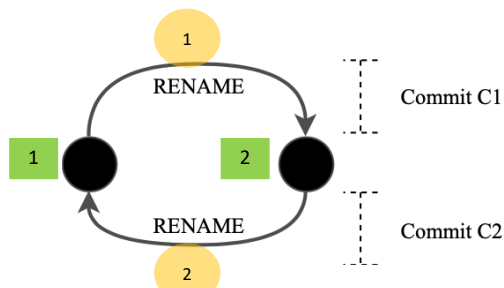
- 本研究はメソッドレベルでの**8つのリファクタリング操作**に集中されている

リファクタリング部分グラフの例



a) 5つの頂点, 4のエッジが持っている部分グラフ

Fig. 6. Example of a refactoring subgraph from MPAndroidChart



b) 2つの頂点, 2つのエッジが持っている部分グラフ

Fig. 10. Example of a refactoring subgraph from Spring Framework

リファクタリング部分グラフの例

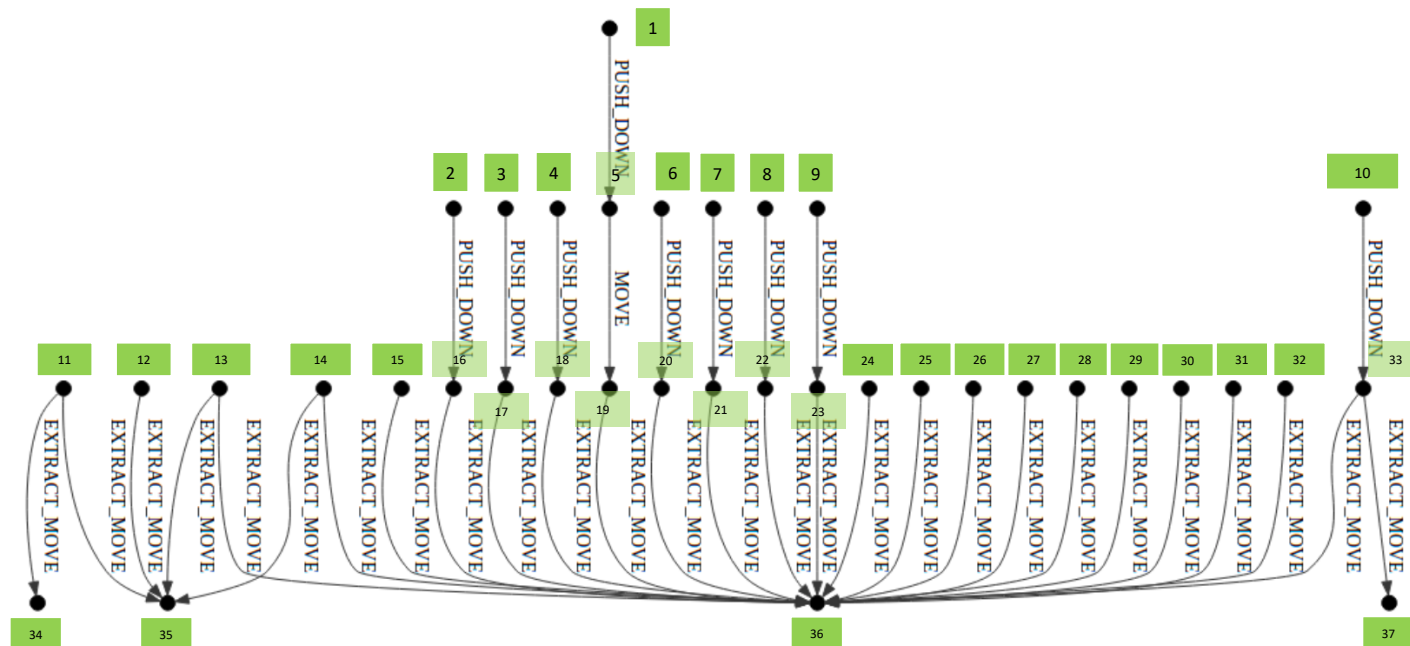


Fig. 13. Example of a large refactoring subgraph from Square Okhttp

c) 37の頂点, 37のエッジが持っている部分グラフ

背景

- ソースコードの保守性と品質を向上させるためにリファクタリングを行う
- リファクタリングによって、新しいソースコードエンティティが生成され、または既存エンティティの構造が変更された
- また、一連のリファクタリングは違う開発者が行うこともある
- このようなりファクタリングは既存手法では検出できない

目的

- メソッドに対する **リファクタリングをグラフ化**することを提案する
- **10個**のソフトウェアプロジェクトから抽出されたリファクタリンググラフの特性を明らかにする
 - リファクタリンググラフの理解が深めるため

アプローチ

1. Javaプロジェクトの選択
2. リファクタリング操作の検出
3. リファクタリンググラフの作成

Javaプロジェクトの選択

TABLE I
SELECTED JAVA PROJECTS

Project	Stars	Forks	Commits	Contributors	Java Files	Latest Version	Description
Elasticsearch	44,489	14,930	48,313	1,273	11,770	7.3.2	Search engine for cloud systems
RxJava	40,622	6,825	5,581	237	1,666	3.0.0-RC3	Library for asynchronous communications
Square Okhttp	34,484	7,521	4,273	189	167	4.2.0	HTTP client
Square Retrofit	33,801	6,254	1,756	129	241	2.6.2	HTTP client
Spring Framework	32,582	21,226	19,752	396	7,203	5.2.0	Framework for web applications
Apache Dubbo	29,353	19,256	3,639	249	1,743	2.7.3	RPC framework
MPAndroidChart	28,647	7,424	2,018	66	220	3.1.0	Library to create charts
Glide	27,289	5,025	2,416	102	647	4.10.0	Library to load images
Lottie Android	26,952	4,278	1,139	76	198	3.0.7	Library to parse animations
Facebook Fresco	15,870	3,595	2,158	170	985	2.0.0	Library to display images

- 10個の有名なJavaプロジェクトが選択された

リファクタリング操作の検出

- リファクタリンググラフの作成に必要なリファクタリング操作を検出する
- refdiffコマンドが使用された

リファクタリンググラフの作成

Algorithm 1: Building refactoring graphs

Input: R (list of refactorings from a system S)

Output: DG (refactoring graph)

```
1 begin  
2    $DG \leftarrow \emptyset, V \leftarrow \emptyset, E \leftarrow \emptyset$   
3   for  $(m1, m2, t) \in R$  do  
4      $V \leftarrow V \cup \{m1, m2\}$   
5      $E \leftarrow E \cup (m1, m2, t)$   
6   end  
7   return  $(V, E)$   
8 end
```

- アルゴリズム1を利用し, リファクタリンググラフを作成した

リファクタリンググラフの作成

TABLE II
FREQUENCY OF REFACTORING SUBGRAPHS

Project	Refactoring Subgraphs				
	All	$len = 1$	%	$len \geq 2$	%
Elasticsearch	2,073	1,934	93.3	139	6.7
RxJava	1,073	975	90.9	98	9.1
Square Okhttp	635	548	86.3	87	13.7
Square Retrofit	169	135	79.9	34	20.1
Spring Framework	3,104	2,604	83.9	500	16.1
Apache Dubbo	483	453	93.8	30	6.2
MPAndroidChart	454	381	83.9	73	16.1
Glide	425	294	69.2	131	30.8
Lottie Android	196	173	88.3	23	11.7
Facebook Fresco	314	279	88.9	35	11.1
Total	8,926	7,776	87.1	1,150	12.9

- 分析されたシステムにおける部分グラフのリファクタリングの頻度（表2）
- 合計8,926個のリファクタリング部分グラフが検出された
- 2つ以上のコミットで実行されたリファクタリングは12.9%
- 時間の経過に伴うリファクタリングを表すものであるため, コミット数が2以上の1,150のリファクタリング部分グラフだけが分析された

リサーチクエスチョン

- RQ1 : リファクタリング部分グラフのサイズは?
- RQ2 : リファクタリング部分グラフにはいくつのコミットで作成されたか?
- RQ3 : リファクタリング部分グラフはどのぐらいの期間にわたって変更がされたか?
- RQ4 : どちらのリファクタリング操作でリファクタリング部分グラフが構成されているか?
- RQ5 : リファクタリング部分グラフは, 同じまたは複数の開発者によって作成されたか?

RQ1 : リファクタリング部分グラフのサイズは?

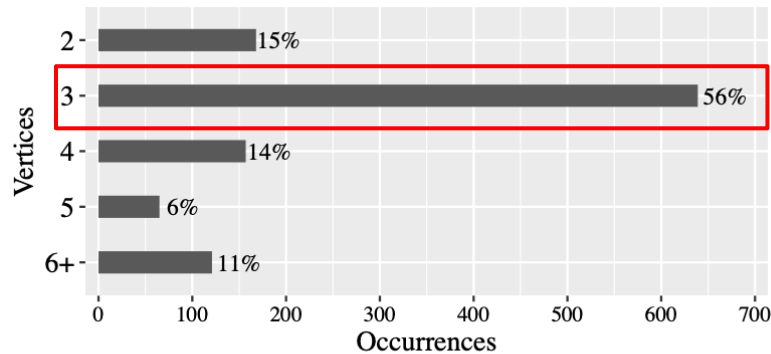


Fig. 4. Number of vertices by refactoring subgraph

ほとんどのリファクタリング部分グラフには3つの頂点がある (639回出現、56%)

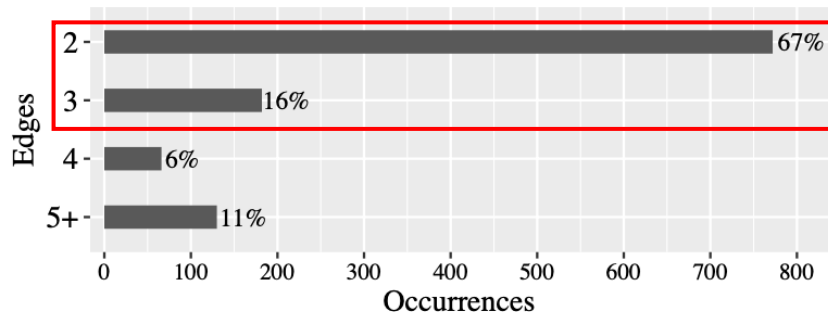


Fig. 5. Number of edges by refactoring subgraph

ほとんどの部分グラフには2つのエッジ (67%) または3つのエッジ (16%) がある

RQ1：リファクタリング部分グラフのサイズは？

- ほとんどのリファクタリング部分グラフは小さい
- 1,150サンプルのうち、ほとんどの部分グラフは2~4（85%）の頂点がある
- また、エッジの数はほとんど2つ~3つ（83%）ということがわかった

RQ2 : リファクタリング部分グラフにはいくつかのコミットで作成されたか?

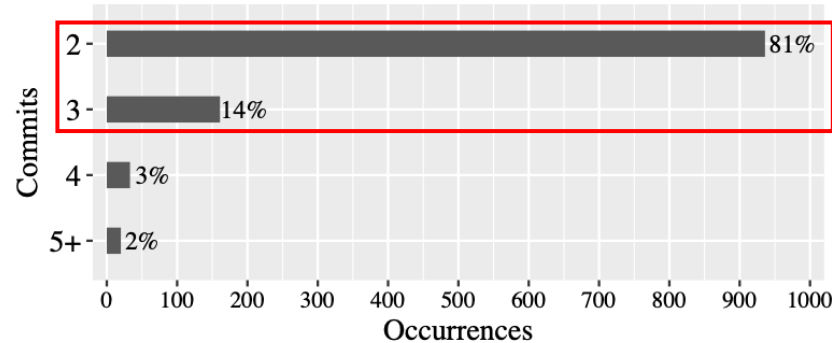


Fig. 7. Number of commits by refactoring subgraph

- 部分グラフごとのコミット数を調査する
- ほとんどのリファクタリング部分グラフは2つのコミット (81%) または3つのコミット (14%) で作成された

RQ3 : リファクタリング部分グラフはどのぐらいの期間にわたって変更がされたか?

- リファクタリング部分グラフの最新コミットと最も古いコミットの間の日数を計算
- 結果は図9で表示
- リファクタリング部分グラフの期間はプロジェクトによって異なることがわかった
- 数日あるものもあるが、ほとんどの部分グラフには数週間または数か月かかる

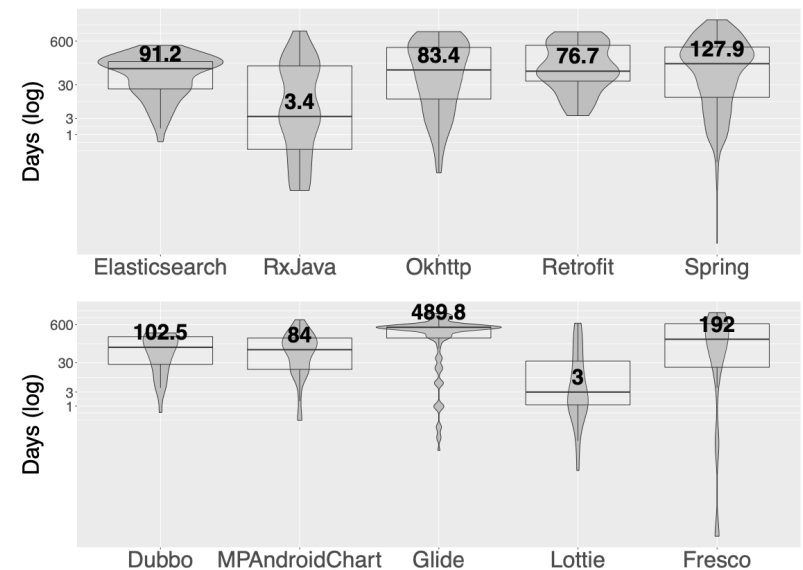


Fig. 9. Age of the refactoring subgraphs

RQ4 : どちらのリファクタリング操作でリファクタリング部分グラフが構成されているか?

TABLE III
FREQUENCY OF REFACTORING OPERATIONS

Refactoring	Occurrences	%
Rename	757	21
Extract and move	685	19
Extract	635	17
Move	579	16
Inline	474	13
Pull up	330	9
Push down	142	4
Move and rename	83	2
All	3,685	100

- まず, 1,150のリファクタリング部分グラフのサンプルでよく使われているリファクタリング操作を示す (表3)
- 次, 部分グラフを2つのグループに分類し, 同種および異種
- 同種グループ: 単一のリファクタリング操作を持つ部分グラフ
- 異種グループ: 少なくとも2つの異なるリファクタリング操作を持つ部分グラフ

RQ4 : どちらのリファクタリング操作でリファクタリング部分グラフが構成されているか?

TABLE IV
HOMOGENEOUS VS HETEROGENEOUS REFACTORING SUBGRAPHS

Project	Homogeneous	%	Heterogeneous	%
Elasticsearch	43	30.9	96	69.1
RxJava	56	57.1	42	42.9
Square Okhttp	22	25.3	65	74.7
Square Retrofit	12	35.3	22	64.7
Spring Framework	138	27.6	362	72.4
Apache Dubbo	6	20.0	24	80.0
MPAndroidChart	16	21.9	57	78.1
Glide	19	14.5	112	85.5
Lottie Android	5	21.7	18	78.3
Facebook Fresco	6	17.1	29	82.9
All	323	28.1	827	71.9

- リファクタリング部分グラフの28%は同種であるが、72%は異種
- ほとんどのリファクタリング部分グラフは異種（71.9%）で、複数のリファクタリング操作が含まれている

RQ4 : どちらのリファクタリング操作でリファクタリング部分グラフが構成されているか?

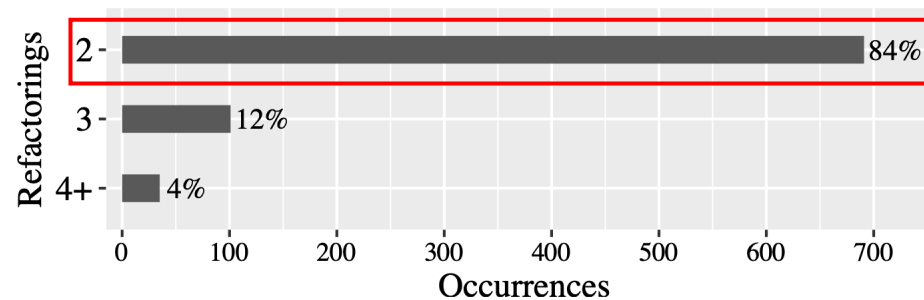


Fig. 11. Number of distinct refactoring operations in heterogeneous subgraphs

- 異種部分グラフに含まれているリファクタリング操作を調査
- 多くの場合, **2つの異なるリファクタリング操作 (84%)** が含まれ, **12%には3つ, 4%に4つ以上の異なる** リファクタリング操作がある

RQ5 : リファクタリング部分グラフは, 同じ
または複数の開発者によって作成されたか?

TABLE V
DEVELOPERS OF REFACTORING GRAPHS

Project	Single dev.	%	Multiple devs.	%
Elasticsearch	32	23.0	107	77.0
RxJava	88	89.8	10	10.2
Square Okhttp	32	36.8	55	63.2
Square Retrofit	14	41.2	20	58.8
Spring Framework	303	60.6	197	39.4
Apache Dubbo	17	56.7	13	43.3
MPAndroidChart	70	95.9	3	4.1
Glide	116	88.5	15	11.5
Lottie Android	11	47.8	12	52.2
Facebook Fresco	10	28.6	25	71.4
All	693	60.3	457	39.7

- リファクタリング部分グラフを2つのグループに分け, 単一の開発者と複数の開発者
- ほとんどのリファクタリング部分グラフは1人の開発者によって作成されている (60%)

まとめ

- リファクタリンググラフを提案し, 10の有名なJavaシステムの分析
- リファクタリンググラフの特性を評価するために5つのリサーチクエスチョンが調査された (サイズ, コミット, 期間, 構成, 開発者)
 - リファクタリング部分グラフの大部分は小さい
 - ほとんどのリファクタリング部分グラフには, 最大3つのコミット (95%) で作成される
 - リファクタリング部分グラフは, 数日から数か月にわたって変更された
 - 多くの場合, リファクタリンググラフは異種グループ (いくつかのリファクタリング操作が含まれている)
 - リファクタリンググラフは, ほとんどが1人の開発者 (60%) によって作成される

所感

- リファクタリングのことがもっと理解できた