



Title	オープンソースソフトウェアにおけるソフトウェアライセンス間の包含関係の調査
Author(s)	眞鍋, 雄貴; 井上, 克郎
Citation	信学技報. 2013, 112(373), p. 61-66
Version Type	VoR
URL	<a href="https://hdl.handle.net/11094/26842">https://hdl.handle.net/11094/26842</a>
rights	copyright©2013 IEICE
Note	

*The University of Osaka Institutional Knowledge Archive : OUKA*

<https://ir.library.osaka-u.ac.jp/>

The University of Osaka

## オープンソースソフトウェアにおける ソフトウェアライセンス間の包含関係の調査

眞鍋 雄貴<sup>†</sup> 井上 克郎<sup>†</sup>

† 大阪大学大学院情報科学研究科 〒 565-0871 大阪府吹田市山田丘 1 番 5 号

E-mail: †{y-manabe, inoue}@ist.osaka-u.ac.jp

あらまし 大規模なソフトウェアを効率よく作成するため、ソースファイルやライブラリなどを再利用することがある。ソースファイルやライブラリを再利用して構築したソフトウェアのライセンスはそれらのソースファイルやライブラリのライセンスを満たすものでなくてはならない。そこで、そのようなライセンスを容易に選択できるよう、既存のオープンソースソフトウェアの集合である Fedora17 のソースパッケージからライセンス間の包含関係を抽出した。結果として、多数のパッケージに登場する包含関係を抽出でき、また、それらが連結することがわかった。また、抽出した包含関係によるライセンス間の到達可能性を調査し、ライセンスの矛盾が起こりうることを実例に基づき示した。

**キーワード** ソフトウェアの再利用、ソフトウェアライセンス、ソフトウェアパッケージ、オープンソースソフトウェア、包含関係

## Empirical Study on Inclusion Relation among Software Licenses of Open Source Software

Yuki MANABE<sup>†</sup> and Katsuro INOUE<sup>†</sup>

† Department of Computer Science Graduate School of Information Science and Technology, Osaka University 1-5 Yamadaoka, Suita, Osaka 565-0871 Japan

E-mail: †{y-manabe, inoue}@ist.osaka-u.ac.jp

**Abstract** Developers reuse source code files in other project and library to develop large software efficiently. A license of the software must meet their licenses. In this paper, we extract inclusion order among software licenses from open source software. We conducted empirical study using a set of source packages in Fedora 17 as a set of open source software. The result shows that we can extract inclusion order from source package and we can connect these relations. In addition, based on these inclusion relations, we also show that license inconsistency may occur.

**Key words** Reuse, Software license, Software package, Open source software, Partial Order

### 1. はじめに

開発コスト削減の一手段として、既存のソフトウェアの一部や全部を、部品として同一システム内や他のシステムで再利用することが広く行われている[1]。近年では、SourceForge.net[2]等のサイトを通じて、ソースファイルやライブラリを容易に取得することが出来る。

一方で、ソースファイルやライブラリを再利用するためには、それらのソフトウェアライセンス（以降、単にライセンス）を確認し、遵守しなければならない。ライセンスとは、著者が定めた著作物の利用の許諾と、許諾を得るための条件である。オープンソースソフトウェアの場合、著者が成果物に対してライセンスを設定している場合、開発者は指定されたライセン

スを遵守することを条件に、オープンソースソフトウェアの改変や、再配布を許諾するという構造になっているのが一般的である[3]。

様々なソースファイルやライブラリを利用して、ソフトウェアを作成したとき、それらのライセンスをすべて満たさなければならない[4]。たとえば、図1のように、自分が作成したソースファイルに加え、他のプロジェクトから取得したソースファイルや2種類のライブラリを再利用してソフトウェアを構築するとする。この時、ソフトウェアは License A, B, C, D の下で配布される成果物から構成されるため、ソフトウェア自体のライセンスはこれらのライセンスをすべて満たさなければならぬ。実際のオープンソースソフトウェアでの例として、Scacchiら[5]は、Google Chrome[6]を調査し、Google Chrome は 14

種類のライセンスの下で配布される 27 種の外部コンポーネントやライブラリを利用しているとしている。

ライセンスは契約文書であるため、複数のライセンスと矛盾しないライセンスを一から開発者が作成するのは容易ではない。そのため、既存のライセンスから選択することが望ましい。しかしながら、多くのライセンスが存在するため、正しく選択するのに多くの労力が必要となる。たとえば、Open Source Initiative に承認された、オープンソースソフトウェアの定義を満たすライセンスだけでも 69 種類存在する[7]。また、BlackDuck は自社が所有する Black Duck KnowledgeBase には 2200 種以上のライセンスに対する詳細なデータが含まれているとしている[8]。

既存研究として、ライセンス間の不整合を扱った研究として、Alspaugh らはライセンスの条項を  $\langle \text{actor}, \text{modality}, \text{action}, \text{object}, \text{license} \rangle$  のタプルで表現し、ソフトウェアトレーサビリティツールの ArchStudio4 上で権利や義務の衝突を計算する手法を提案した[9]。German らはライセンスの不整合が生じたときの対処をライセンス統合パターンとして整理した[4]しかし、不整合が分かったとしても、それを解消するのは容易ではなく、また、代替となるライセンスを見つけるのは難しい。また、従来のライセンス毎の制約の強さを分類した表現として、ライセンスのコピー権限性に着目した分類がある[10]。この分類では、同一のカテゴリ内に複数のライセンスが分類されており、そのライセンス間での制約の強さを比較していない。

そこで、本研究では、ソフトウェアのライセンスとソースファイルのライセンス間の関係を包含関係とみなし、既存のオープンソースソフトウェアからライセンスの包含関係を抽出することによって、矛盾しないライセンスを開発者が容易に選択できるようにする。ソフトウェア全体のライセンスは各ソースファイルのライセンスを満たしていると考えられるため、ソフトウェアのライセンスはソースファイルのライセンスを包含しているといえる。本稿では、既存のオープンソースソフトウェアの集合として Fedora 17 におけるソースパッケージに着目し、ライセンスの包含関係を実際に抽出した。その結果、様々なライセンス間において包含関係を見つけることができ、複数のライセンス間で推移できることも示した。一方では、すべてのライセンスから到達可能なライセンスは存在しなかつたため、オープンソースソフトウェアのファイルの再利用によって、ライセンスの矛盾が生じうる可能性があることを実例から示した。

本稿における主な貢献は以下のとおりである。

- 実際のソフトウェアパッケージから包含関係を抽出し、どの様にライセンス間に包含関係があるかを示した。
- 抽出したライセンス間の包含関係を用いて、ライセンス間の到達可能性を調査した結果、全てのライセンスから到達可能なライセンスは存在しないことから、オープンソースソフトウェアを再利用した際にライセンスの矛盾が生じうる可能性を示した。

## 2. 用語

ソースファイルはコメントとプログラムコードを含んでいる。

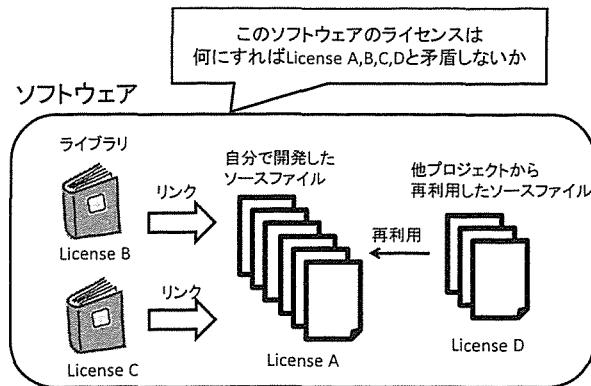


図 1 本稿が想定する問題

コメントはプログラムコードの説明や、ライセンスの指示、その他の目的のために使用されている。ライセンスの指示は一般的に複数の自然言語の文からなっている。

ソースパッケージとは、管理され、インストール方法が提供されている、ソフトウェアのソースファイル集合を含むファイルである。ソースパッケージはソフトウェアのソースファイル集合とその関連情報を記載したファイル、ビルト用のファイルからなる。関連情報としてライセンスが記載されている場合がある。

ソフトウェアライセンス（以下、ライセンス）はソフトウェアの作者により定められたソフトウェアのユーザーへの指示の集合である。ライセンスはソフトウェアプロダクト全体だけでなく、ソフトウェアプロダクトを構成する個々のソースファイルについても決まっている。ライセンスの中には変更を受け、複数の版が存在するものもある。例として、GNU General Public License(GPL)には version1, version2, version3 が存在する。また、GPLの場合、“*GPL version 2 or later*”とソースファイル中のライセンスの指示に記載されている場合、そのソースファイルは version2 以降の最新の GPL が適用されているとすることができる。本稿では、個々のライセンスの法的問題については議論しない。

表 1 に本稿で使用するライセンスの名前と略称を示す。ライセンスが複数の版を持つ場合、特定の版を示すために接尾辞 v<number> を用いる。また、“or later”を接尾辞+で表現する。例として、GPL の version2 or later を GPLv2+と略する。

ライセンス間の包含関係は、あるライセンスで配布されている成果物を別のライセンスで配布できるとき、そのライセンス間にある関係とする。例として、BSD2 は BSD3 から endorsement clause を取り除いたものである。そのため、BSD3 を満たすことで、BSD2 を満たすことができる。この時、包含関係を  $\sqsubseteq$  という記号で表すとすると、 $\text{BSD2} \sqsubseteq \text{BSD3}$  と表現できる。ソースパッケージのライセンスはソースファイルのライセンスを包含していると考えられるため、本稿ではソースファイルのライセンス  $\sqsubseteq$  ソースパッケージのライセンスであるとしている。

ライセンス間の包含関係のグラフとは、ライセンス間の包含関係を有向グラフとして表現したものである。各ライセンスを頂点、包含関係を辺として表す。例として、前述の  $\text{BSD2} \sqsubseteq \text{BSD3}$

例では、BSD2 と BSD3 をそれぞれ頂点とし、BSD2 から BSD3 への有向辺をもつグラフとして表現できる。ライセンス間の包含関係のグラフを用いることで、異なる複数のライセンスに基づくソースファイルやライブラリを含むソフトウェアを記述するために選択できるライセンスを探す問題を、それぞれのライセンスが対応する頂点全てから到達できる頂点を探す問題に帰着できる。

### 3. Empirical Study

本研究では、オープンソースにおけるソフトウェアライセンス間の包含関係とその再利用への適用可能性を調査するため、以下の 2 つのリサーチクエスチョンを設定し、実験を行った。

**RQ1** 既存のソフトウェアから得られる包含関係はどの様な物であるか。

**RQ2** 得られた包含関係から、ライセンス間の矛盾が起きうることを確認できるか。

#### 3.1 方 法

本実験では、調査対象として Linux のディストリビューションの一つである Fedora 17 のミラーサイト<sup>(注1)</sup>から取得した src.rpm ファイル 2132 個を用いた。src.rpm ファイルは、主にソースファイル集合を圧縮したアーカイブと、ライセンス等が記載された SPEC ファイルから構成されている [11]。

本実験では、個々のソースファイルのライセンスを特定するため、Ninka [12] を用いた。Ninka はソースファイルを入力とし、入力されたファイルに対して特定したライセンス名、つまり BSD4 や GPLv2+ と出力する。Ninka は 110 種の異なるライセンスを 93% の精度で特定できる [12]。また、Ninka は対応していないライセンスに対しては特定できない。

本実験では、以下の手順によりライセンス間の包含関係を

表 1 本稿で使用する一般的なオープンソースライセンスとその略称

Abbrev.	Name
Apache	Apache Public License
BSD4	Original BSD, also known as BSD with 4 clauses
BSD3	BSD4 minus advertisement clause
BSD2	BSD3 minus endorsement clause
CPL	Common Public License
CDDLic	Common Development and Distribution License
EPL	Eclipse Public License
GPL	General Public License
LesserGPL	Lesser General Public License (successor of the Library GPL, also known as LGPL)
LibraryGPL	Library General Public License (also known as LGPL)
MIT/X11	Original license of X11 released by the MIT
MITold	License similar to the MIT/X11, but with different wording
MITX11noNotice	MIT/X11 minus notice clause

(注1) : <ftp://ftp.jaist.ac.jp/pub/Linux/Fedora/releases/17/Fedora/source/>

得た。

(1) 各パッケージを解凍し、パッケージごとに SPEC ファイルとアーカイブを取得する。本研究では、tar.gz ファイルをアーカイブとみなした。

(2) 各パッケージの SPEC ファイルを調べ、各ソースパッケージのライセンスを特定する。

(3) 各パッケージのアーカイブを解凍し、ソースファイルを取得する。

(4) 各パッケージ中にあるソースファイルのライセンスを Ninka を用いて特定する。本実験では、C(.c,.h), C++(.cc,.cpp,.cxx), Java(.java), Perl (.pl, .pm), Python(.py) で記述されたソースファイルを使用した。

(5) Ninka でのライセンス特定のための規則や、Fedora でのライセンス名のルール [13] を参考に、Ninka とパッケージでのライセンス名の対応表を作成する。

(6) 特定したライセンスが 1 種類だけのパッケージと、その中に含まれるソースファイルのうち特定されたライセンスが 1 種類のみのソースファイルを特定し、ソースファイルのライセンス → パッケージのライセンスとなるように包含関係を抽出する。その際、ライセンスの名前は作成したライセンス名の対応表を用いて同じライセンスは同じ名前で表すようにした。なお、対応表のみで対応できなかったパッケージは以下の作業を行った。

- 複数のライセンスが対応する記法 (BSD, MIT, GPL, LGPL) については、各パッケージを調査し、ドキュメント等で確認した。確認できなかったものは対象から外した。
- フォント、ドキュメントでのみ用いられるライセンスは除外した。

(7) 得られた関係のうち、閾値以上のパッケージに存在した関係のみを抽出する。閾値は 10 を用いた。これは、使用できたパッケージの 1% 以上が支持している関係を包含関係として抽出するためである。

本実験で使用したパッケージとソースファイルの個数と、そのうち包含関係抽出に用いた個数を表 2 に示す。パッケージ数においては全体の 1/2 程度、ソースファイルにおいては全体の 1/3 程度が包含関係の抽出に利用できた。

包含関係抽出に用いたパッケージとソースファイルの個数をライセンスごとに示したものと表 3 と表 4 に示す。これらの表から、パッケージ、ソースファイルとともに GPLv2+が多いことが分かる。また、EPLv1 のようにソースファイルでは多いが、パッケージにはあまり現れないライセンスも存在していた。

#### 3.2 RQ1

RQ1 に答えるため、10 パッケージ以上で確認できた包含関係を用いて、ライセンス間の包含関係を作成した。作成したグラフを図 2 に示す。その結果、多くの包含関係が連結しているこ

表 2 使用したパッケージとソースファイルの個数

	総計	包含関係抽出に利用
パッケージ数	2132	932
ソースファイル数	458366	144297

とが分かる。一方で、GPLv2+と LibraryGPLv2+, GPLv2, GPLv3との間には両方向の包含関係が存在している。特に、 $\text{GPLv2+} \supseteq \text{LibraryGPLv2+}$ については、従来のコピーレフト性の分類で考えると、現れない包含関係となっている。

以上より、RQ1への答えは、「ソースパッケージから得られた包含関係は多くの部分で連結しているが、一部で従来の分類では考えられない包含関係も含んでいる」とする。

### 3.3 RQ2

RQ2に答えるため、求めたライセンス間の包含関係を用いて、ライセンス間の到達可能性について調べた。到達可能性は、ライセンス間の包含関係を隣接行列として表現したのち、ワーシャル法を用いて求めた[14]。

表3 パッケージのライセンス（10パッケージ以上）

ライセンス名	パッケージ数
GPLv2+	334
LibraryGPLv2+	199
GPLv2	89
BSD3	65
MITX11noNotice	61
Apachev2	57
GPLv3+	56
LibraryGPLv2	39
X11mit	28
EPLv1	23
BSD2	23
LesserGPLv3+	19
X11	15
MITold	15
ZLIB	13
BSD4	12
publicDomain	10

表4 ソースファイルのライセンス（1000ファイル以上）

ライセンス名	ファイル数
EPLv1	39916
GPLv2+	35097
GPLv3+	29813
Apachev2	24549
LesserGPLv2.1+	20677
BSD3	15656
GPLv2	15377
LibraryGPLv2+	8351
MITX11noNotice	7283
boostV1	6336
BSD2	5012
LesserGPLv3+	3313
LesserGPLv2+	3178
FreeType	2822
LesserGPLv2.1	2097
BSD4	1414
artifex	1228
publicDomain	1142
X11mit	1065

10パッケージ以上で見られたライセンス間の包含関係から求めた到達可能性を表6に示す。この表では、行が始点となるライセンスであり、列が終点となるライセンスを示す。a行に対応するライセンスがb列に対応するライセンスに到達可能である時、a行b列の値を1とし、そうでなければ0としている。

この表より、どのライセンスからも到達できるライセンスは存在していないことが分かる。特に、MITX11noNoticeについては、表3と表4より、パッケージのライセンスやソースファイルのライセンスとして多く現れているにもかかわらず、このライセンスへ到達できる包含関係はなかった。そのため、今回得られた包含関係からは、ソースファイルやライブラリの選択によっては、それらのライセンスを満たすライセンスが存在せず、配布が出来ない場合がある可能性があるといえる。

よって、RQ2に対する答えは「Fedora17から得られたライセンス間の包含関係においては、どのライセンスからも到達で

表5 ライセンス名の対応表

パッケージでのライセンス名	Ninkaでのライセンス名
ApacheSoftwareLicense	Apachev2
ASLv2	Apachev2
ASLv1.1	Apachev1.1
BSDwithadvertising	BSD4
BSD-compatible	BSD3
Copyrightonly	NONE
CPL	CPLv1
CDDL	CDDLicV1
EPL	EPLv1
ISC	BSD2
GPLv2	LibraryGPLv2
GPL2.1	LesserGPLv2.1
GPLv3	LesserGPLv3
MPLv1.1	MPLv1.1
PHP	phpLicV3.01
PublicDomain	publicDomain
W3C	W3CLic
zlibLicense	ZLIB
zlib	ZLIB

表6 各ライセンス間の到達可能性

	a	b	c	d	e	f	g	h	i	j	k	l	m
(a) BSD2	1	0	1	1	1	0	0	0	1	1	1	0	0
(b) BSD3	1	1	1	1	1	0	0	0	1	1	1	0	0
(c) GPLv2	0	0	1	1	1	0	0	0	1	1	1	0	0
(d) GPLv2+	0	0	1	1	1	0	0	0	1	1	1	0	0
(e) GPLv3+	0	0	1	1	1	0	0	0	1	1	1	0	0
(f) LesserGPLv2+	0	0	1	1	1	1	0	0	1	1	1	0	0
(g) LesserGPLv2.1	0	0	0	0	0	0	1	0	0	1	0	0	0
(h) LesserGPLv2.1+	0	0	1	1	1	0	0	1	1	1	1	0	0
(i) LesserGPLv3+	0	0	0	0	0	0	0	0	1	0	0	0	0
(j) LibraryGPLv2	0	0	0	0	0	0	0	0	0	1	0	0	0
(k) LibraryGPLv2+	0	0	1	1	1	0	0	0	1	1	1	0	0
(l) MITX11noNotice	0	0	1	1	1	0	0	0	1	1	1	1	0
(m) publicDomain	1	1	1	1	1	0	0	0	1	1	1	1	0

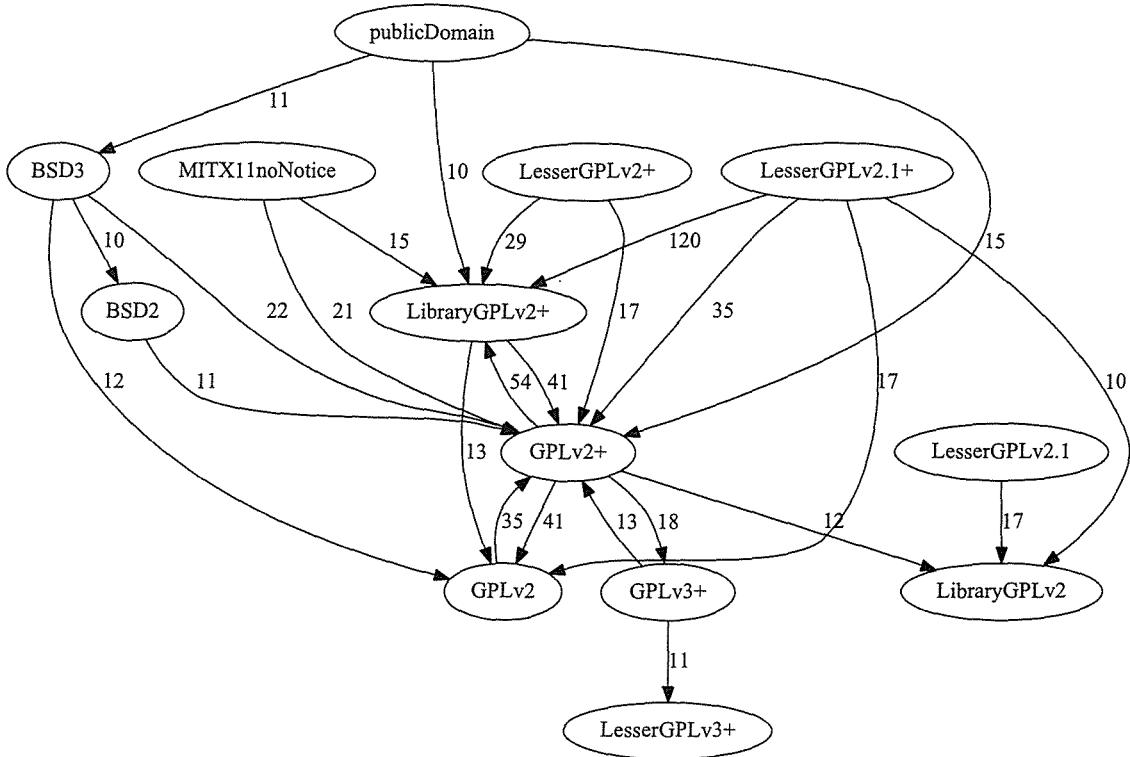


図 2 10 パッケージ以上で確認できたライセンス間の包含関係. 各辺の数値はその包含関係を抽出できたパッケージの数を表す.

きるライセンスはないため, ライセンスの矛盾が起こりうる.」とする.

#### 4. 制限

本稿では, ソースファイルの利用方法を考慮できておらず, パッケージとそれに含まれるソースファイル間の関係のみ見ている. パッケージにソースファイルがあったとしても, 実際ににはソフトウェアには含まれない場合がある. 例として, テストケース用のファイルや, ビルド用のスクリプト等がある. そのため, そのパッケージから抽出した包含関係が実態と異なる場合がある. しかしながら, 包含関係の抽出の際に閾値を設けているため, この影響は小さいと考える. 今後, 各ソースファイルの利用方法を調査し, 包含関係を抽出する際に考慮することによって包含関係を洗練したい.

Fedora17 から得たソースパッケージが偏っている場合, 抽出した包含関係の傾向に影響を与える可能性がある. 表 3 にある通り, GPLv2+となるパッケージが非常に多かった. そのため, GPLv2+に関する部分の包含関係は多く取れているが, 一方で少数のパッケージで出ている包含関係が抽出できなくなっている可能性がある. しかしながら, 取得できた包含関係については変更されることはないため, 問題ないと考える. 今後, SourceForge.net などの他のソフトウェア毎のライセンスが分かれるソフトウェア集合から取得したソフトウェアを用いて同様の実験をすることによって, 今回取得できなかった包含関係を取得できるのではないかと考える.

Ninka から得られた結果が誤っている場合, 抽出した包含関係

に誤りがある可能性がある. Daniel et.al. [12] によると, Ninka はライセンスを特定できたソースファイルのうち, 96.6%に対し, 正しくライセンスを特定できている. 我々はこれは十分に高く, そのため, Ninka のライセンス特定の誤りは抽出した包含関係に影響を与えないと思われる.

ソースパッケージのライセンスが誤って SPEC ファイルに記載している場合, 抽出した包含関係に誤りがある可能性がある. Di Penta ら [15] の研究においても, ソースパッケージのライセンスがソースファイルのライセンスと不整合を起こしていることが示されている. 本研究では, 10 以上のパッケージに現れる包含関係のみ抽出しており, 1 つのソースパッケージのライセンスが誤っていたとしても包含関係が変更されることはないと考える.

#### 5. 関連研究

ソースパッケージとライセンスの関係について調査した研究として, Di Penta ら [15] の研究がある. Di Penta らは, ソフトウェアパッケージにおけるライセンスの整合性について調査している. 本研究では, ソースパッケージのライセンスとそれに含まれるソースファイルのライセンスを特定し, ソースパッケージのライセンスがソースファイルのライセンスより古いものであったり, ソースファイルのライセンスの変更をソースパッケージのライセンスに反映されていない等の問題を指摘した. 本研究では, ソフトウェアパッケージ内でのライセンスを調査している点で, Di Penta らの研究と類似している. しかし, 本研究では, ソースパッケージのライセンスとソースファ

イルのライセンス間の関係だけでなく、それらの推移性も調査している。

ライセンスがプロジェクトに与える影響に関する研究として、Colazo ら [16], Stewart ら [17] の研究がある。Colazo らはコピーレフト性のあるオープンソースライセンスを用いるオープンソースソフトウェアプロジェクトはコピーレフト性のないオープンソースライセンスを用いるプロジェクトと比べ、開発者の地位やコーディングの活発さ、永続性が高く、また、開発期間が短くなると述べている。Stewart らはオープンソースライセンスの制約性と組織的資金援助がオープンソースソフトウェア開発へのユーザの興味や、開発者の活動へどう影響を与えるか調べた。その結果として、ユーザは非商業組織により資金援助され、非制約的ライセンスを用いるプロジェクトに最もひきつけられるとした。本稿で求めた包含関係があることで、ライセンス間の矛盾を回避し、より戦略的にライセンスを選択できるようになるのではないかと考える。

Xia ら [18] は、入力したソースファイルからクエリを生成し、ソースファイル検索システムに入力することで類似ファイルを取得し、ウェブ上にあるソースファイルからコードクローンを検出するというツールを提案している。このツールの評価では、入力したファイルと重複するソースコードを持つファイルがどのようなライセンスであるのかを調べている。本研究の成果は、このライセンスを確認し、矛盾していないかを調査するのに役立つと考えられる。

## 6. まとめと今後の課題

本稿では、複数のライセンスと矛盾しないライセンスの選択を開発者にとって容易なものとするため、ライセンスの包含関係を Fedora17 に含まれるソースパッケージから抽出した。その結果、様々なライセンス間において、包含関係を発見できた。一方で、すべてのライセンスから到達できるライセンスは存在せず、ライセンスの矛盾が起こりうることを確認した。

今後の課題として、制限に挙げたソースファイルの利用方法を考慮した包含関係の洗練がある。また、今後は、この包含関係に基づき、各ライセンスの条文の解析等も行っていきたい。

**謝辞** 本研究は、日本学術振興会研究費補助金若手研究（B）（課題番号:24700029）の助成を得た。また、インターン生として本研究の初期段階に協力して頂きました大林菜々氏に深く感謝いたします。

## 文 献

- [1] C.L. Braun, “再利用,” ソフトウェア工学大辞典, ed. by J.J. Marciniaik, 第 1 卷, pp.338–405, 朝倉書店, 1994.
- [2] “Sourceforge.net”, <http://sourceforge.net/>.
- [3] “ビジネスニュースにおけるオープンソースソフトウェアの法的リスクに関する調査”. <http://www.ipa.go.jp/about/jigyoseika/04fy-pro/open/2004-741d.pdf>.
- [4] D.M. German and A.E. Hassan, “License integration patterns: Addressing license mismatches in component-based development,” Proceedings of 31st International Conference on Software Engineering, pp.188–198, 2009.
- [5] W. Scacchi and T.A. Alspaugh, “Understanding the role of licenses and evolution in open architecture software ecosystems,” Journal of Systems and Software, vol.85, no.7, pp.1479–1494, 2012.
- [6] Google, “Google chrome”. <https://www.google.com/intl/en/chrome/browser/>.
- [7] Open Source Initiative, “Open source licenses”. <http://opensource.org/licenses/index.html>.
- [8] “Black duck knowledgebase”. <http://www.blackducksoftware.com/products/knowledgebase>.
- [9] T.A. Alspaugh, H.U. Asuncion, and W. Scacchi, “Intellectual property rights requirements for heterogeneously-licensed systems,” Proceedings of 17th IEEE International Requirements Engineering Conference, pp.24–33, Sept. 2009.
- [10] M. Kechagia, D. Spinellis, and S. Androusellis-Theotokis, “Open source licensing across package dependencies,” Proceedings of 14th Panhellenic Conference on Informatics (PCI), pp.27–32, Sept. 2010.
- [11] T. ‘spot’ Callaway, “Packaging:guidelines”. [https://fedoraproject.org/wiki/Packaging:Guidelines#Writing\\_a\\_package\\_from\\_scratch](https://fedoraproject.org/wiki/Packaging:Guidelines#Writing_a_package_from_scratch).
- [12] D.M. German, Y. Manabe, and K. Inoue, “A sentence-matching method for automatic license identification of source code files,” Proceedings of 25th IEEE/ACM International Conference on Automated Software Engineering, pp.437–446, 2010.
- [13] T. ‘spot’ Callaway, “Licensing:main - fedoraproject”. <https://fedoraproject.org/wiki/Licensing:Main>.
- [14] S. Warshall, “A theorem on boolean matrices,” J. ACM, vol.9, no.1, pp.11–12, Jan. 1962.
- [15] D.M. German, M. Di Penta, and J. Davies, “Understanding and auditing the licensing of open source software distributions,” Proceedings of 18th IEEE International Conference on Program Comprehension, pp.84–93, Braga, Portugal, 2010.
- [16] J. Colazo and Y. Fang, “Impact of license choice on open source software development activity,” Journal of the American Society for Information Science and Technology, vol.60, no.5, pp.997–1011, 2009.
- [17] K.J. Stewart, A.P. Ammeter, and L.M. Maruping, “Impacts of license choice and organizational sponsorship on user interest and development activity in open source software projects,” Info. Sys. Research, vol.17, pp.126–144, June 2006.
- [18] P. Xia, Y. Manabe, N. Yoshida, and K. Inoue, “Development of a code clone search tool for open source repositories,” コンピュータソフトウェア, vol.29, no.3, pp.181–187, Aug. 2012.