



HAL
open science

Knowledge Extraction from Description Logic Terminologies

Jieying Chen

► **To cite this version:**

Jieying Chen. Knowledge Extraction from Description Logic Terminologies. Information Retrieval [cs.IR]. Université Paris-Sud, 2018. English. NNT : . tel-04471349

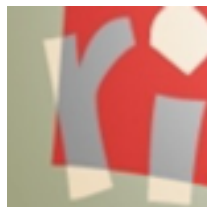
HAL Id: tel-04471349

<https://theses.hal.science/tel-04471349>

Submitted on 21 Feb 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Knowledge Extraction from Description Logic Terminologies

Thèse de doctorat de l'Université Paris-Saclay
préparée à l'Université Paris-Sud

École doctorale n°580 Sciences et Technologies de l'Information et de
la Communication (STIC)
Spécialité de doctorat : Informatique

Thèse présentée et soutenue à Gif-sur-Yvette, le 30 novembre 2018, par

Jieying CHEN

Composition du Jury :

Anne VILNAT Professeure, Université Paris-Saclay	Présidente
Marie-Christine ROUSSET Professeure, Université Grenoble-Alpes	Rapporteuse
Thomas SCHNEIDER Professeur, Université de Brême	Rapporteur
Marie-Laure MUGNIER Professeure, Université de Montpellier	Examinatrice
Yue MA Maître de Conférences, Université Paris-Saclay	Encadrante
Philippe DAGUE Professeur, Université Paris-Saclay	Directeur de thèse

Acknowledgements

I would like to express my special appreciation and thanks to the director of my thesis Professor Philippe Dague. I am always surprised by how fast that you reply my emails, even when I was not your student. You are always very supportive and make sure that my thesis was on the right track. Besides, you are always there to help me when I had some administrative issues with the university and financial problems for attending conferences and academic visits. I have learned a lot from you, how to organize things and how you treat other people nicely. I am extremely lucky that I have such an amazing supervisor during my PhD.

I would especially like to thank Michel and Dirk. By working with you, I learned how to do serious research, how to write theoretical texts, how to do research on difficult topics, and what high-standard research is. I truly appreciate your generous efforts and help me better understand theory of logical difference. To be honest, at the beginning of our collaboration, I was complaining that how picky you are, but I understand you much better later: this work is so complicated that you have to be picky and extremely careful. Any tiny mistake will introduce a big error to the work.

I thank the jury members of my thesis: Professor Marie-Christine Rousset and Professor Thomas Schneider gave me a lot of useful comments for improving my thesis after dedicating to read my thesis carefully; Professor Marie-Laure Mugnier inspired me some idea for the future work in the defense; and Professor Anne Vilnat accepted to be the president of the jury and provided me useful documents timely for my visa application.

I would also like to thank all members in the team LaHDAK for your support. I thank Joe, Sanjay and Luis for the happy time we spent together. You made my life shining in Paris. I also thank Ismail, Patrick, Theo and Andreas for giving me very nice suggestion and useful information.

Special thanks to my boyfriend. Thank you so much for always being there when I am happy or sad, smiling or crying. It is a bit dramatic how we have been together and what we have been through. I am very grateful that we are together now and you care about me so much.

好的，现在开始中文部分。首先，当然要谢谢亲爱的可爱的跃姐。五年前，我们相识于德累斯顿。在遇到你之前，我脑子对女博士还是那种刻板的印象，觉得做科研是件很无聊和孤独的事儿。遇见你之后，看到你那时候在德国的状态，我改变了原来的想法。原来女博士也不是传说中的灭绝师太，原来跟大家一起做科研是一件多么有趣的事儿。然后呢，我就想开始读博啦，想继续做best excerpts, 也想在社会中多个跟大家印象中不一样的女博士的例子:) 在这五年中，你是我的导师，我们一起在白板上讨论问题，写论文，赶deadline。你也是我的朋友，在生活中给我很多的帮助和鼓励。如果当时我们没有遇见，也就不会有这一篇博士论文了。

其次，我想谢谢在LRI认识的朋友，谢谢许钊，王战，张正(虽然你是Limsi的^^), 王伟，Lina Ye, 史鹤欢, Kevin, 还有其他中国朋友那姐，张曦予。因为有你们，就算在国外，我也不觉得孤单。

谢谢我硕士期间在吉大的导师刘磊教授。如果没有您当初的允许和支持，让我到德国进行交换学习，或许我今天也不会去欧洲读博了。谢谢张鹏师哥在我出国之前，在我犹豫迷茫时对我的指点。同时谢谢张睿师哥带我进入了语义网的大门，在硕士期间对我的帮助。

最后我要谢谢我最爱的爸爸妈妈，谢谢你们让我养成了独立自主的个性，谢谢你们总能让我无后顾之忧地做自己想做的事儿。你们永远是我最强后盾！

Abstract

An increasing number of large ontologies are being developed and made available, e.g., in repositories such as the NCBO Bioportal. Ensuring access to the knowledge contained in ontologies that is most relevant to users has been identified as an important challenge. In this work, we tackle this challenge by proposing three different approaches to extracting knowledge from Description Logic ontologies: extracting minimal ontology modules (i.e., sub-ontologies that are minimal w.r.t. set inclusion while still preserving all entailments over a given vocabulary); computing best ontology excerpts (a certain, small number of axioms that best capture the knowledge about the vocabulary while allowing for a degree of semantic loss); and determining projection modules (sub-ontologies of a target ontology that entails subsumption, instance or conjunctive queries that follow from a reference ontology). For each of these approaches, we are interested in extracting not only one but all instances of the module notion. For computing minimal modules and best excerpts, we introduce the notion of subsumption justification as a generalisation of the notion of a justification (a minimal set of axioms needed to preserve a given logical consequence) to capture the subsumption knowledge over the vocabulary. Similarly, for computing projection modules, we introduce the notion of projection justifications that preserve the answers to one of three query types as given by a reference ontology.

We develop algorithms for computing subsumption (projection) justifications that collect all axioms that are relevant for answering the queries. The process of extracting minimal sub-ontologies is guided by specific simulation notions between ontologies. Different algorithms are required for computing subsumption justifications and projection justifications for the different query languages. The algorithm for computing subsumption justifications collects all minimal sub-ontologies that maintain the simulation in order to preserve the answers to subsumption queries. Similarly, the algorithm for computing projection justifications also employs the simulation notion, but it collects the relevant axioms in a target ontology that preserve the same answer to different types of queries in the reference ontology.

Finally, we evaluate our approaches using a prototype implementation of the algorithms on large ontologies.

Résumé

Un nombre croissant d'ontologies de grande taille sont en cours de développement et rendues disponibles, par exemple dans les référentiels tels que le bioportail de NCBO. Garantir l'accès aux connaissances contenues dans des ontologies qui sont les plus pertinentes pour les utilisateurs est un défi important. Dans ce travail, nous abordons ce défi en proposant trois approches différentes pour extraire des connaissances des ontologies en logique de description :

- Extraire les modules minimaux d'une ontologie (c'est-à-dire les sous-ontologies minimales pour l'inclusion ensembliste qui préservent toutes les implications sur un vocabulaire donné). Les modules peuvent aider à comprendre la structure interne d'une ontologie. Par conséquent, l'extraction des modules minimaux d'une ontologie peut être utile dans de nombreux cas. Par exemple, le raisonnement à base d'ontologies, la visualisation d'ontologies, la réutilisation d'ontologies, la structuration de connaissances.
- Calculer les meilleurs extraits d'une ontologie (un certain nombre, petit, d'axiomes qui capturent au mieux la connaissance du vocabulaire tout en permettant un certain degré de perte sémantique). Ceci afin de résoudre le problème de l'absence de contrôle du nombre d'axiomes contenus dans un module minimal. Même des modules minimaux pour de petites signatures peuvent être volumineux, ce qui rend la compréhension humaine difficile. Par contre, un meilleur extrait préserve autant de connaissances que possibles relativement à une taille donnée.
- Déterminer les modules de projection (sous-ontologies d'une ontologie cible qui impliquent des requêtes de souscription, d'instance ou conjonctives qui découlent d'une ontologie de référence). Différent des notions précédentes, le module de projection prend en compte une seconde ontologie comme la référence et le but est d'extraire une sous-ontologie de l'ontologie cible qui préserve les connaissances sur un vocabulaire donné impliquées par cette ontologie de référence.

Pour chacune de ces approches, nous souhaitons extraire non seulement une, mais toutes les instances de la notion correspondante de module. Pour le calcul des modules minimaux et des meilleurs extraits, nous introduisons la notion de justification de la subsomption en tant que généralisation de la notion de justification (un ensemble minimal d'axiomes nécessaires pour préserver une conséquence logique donnée) afin de capturer la connaissance déduite par subsomption sur le vocabulaire. De même, pour le calcul des modules de projection, nous introduisons la notion de justification de projection qui préserve les réponses à l'un des trois types de requête impliquées par une ontologie de référence.

Nous développons des algorithmes pour calculer les justifications de subsomption (resp. de projection) qui rassemblent tous les axiomes pertinents pour répondre aux requêtes. Le processus d'extraction de sous-ontologies minimales est guidé par des notions de simulation spécifiques entre ontologies. Différents algorithmes sont nécessaires pour calculer les justifications de subsomption et de projection pour les différents langages de requêtes. L'algorithme de calcul des justifications de subsomption recueille toutes les sous-ontologies minimales qui respectent les règles de simulation afin de conserver les réponses aux requêtes de subsomption. De même, l'algorithme de calcul des justifications de projection utilise également la notion de simulation, mais il recueille les axiomes pertinents dans une ontologie cible qui conservent la même réponse à différents types de requêtes dans l'ontologie de référence.

Nous évaluons nos approches à l'aide d'un prototype d'implémentation des algorithmes sur les grandes ontologies, y compris SNOMED CT et NCI. Les évaluations montrent qu'il est possible de calculer tous les modules minimaux, les meilleurs extraits, et les modules de projection d'ontologies à grande échelle. En plus, les modules minimaux et les meilleurs extraits peuvent être beaucoup plus petits que les ontologies originales, ce qui montre l'avantage de ces notions.

Contents

1	Introduction	1
1.1	Ontology Modularity	2
1.2	Ontology Excerpts	4
1.3	Projection Modules	6
1.4	Structure of the Thesis	9
2	Preliminaries	13
2.1	Description Logics	13
2.2	Logical Difference between Ontologies	18
2.2.1	Proof system for \mathcal{ELH}^r	23
2.2.2	Properties Regarding Logical Differences	25
2.3	Ontology Modules	27
2.3.1	Inseparability and Semantic Modules	27
2.3.2	Locality-Based Modules	29
2.4	Ontology Justification	30
2.5	Partial Max-SAT Problem	31
3	Extraction of Ontology Subsumption Justifications	35
3.1	Subsumption Justifications of \mathcal{ELH} -Terminologies	35
3.1.1	Computing Role Subsumption Justifications	37
3.1.2	Computing Subsumer Justifications	37
3.1.3	Computing Subsumee Justifications	39
3.1.4	Proofs	45
3.2	Subsumption Justifications of \mathcal{ELH}^r -Terminologies	53
3.2.1	Computing Subsumer Justifications	54
3.2.2	Computing Subsumee Justifications	56
3.3	Evaluation	59

4	Extraction of Ontology Subsumption Modules	63
4.1	Definition of Subsumption Module	63
4.2	Black-Box Algorithm	66
4.2.1	Computing a Single Minimal Module	66
4.2.2	Computing All Minimal Modules	69
4.3	Glass-Box Algorithm	73
4.4	Evaluation	73
5	Extraction of Best Ontology Excerpts	77
5.1	Ontology Excerpts	77
5.2	Encoding Approach of Computing Best Excerpt	82
5.3	Computing Best Excerpt for Weighted Signatures	87
5.4	Ranking Axioms in Best Excerpts	91
5.5	Evaluation	92
6	Extraction of Ontology Projection Module	95
6.1	Projection Modules	95
6.2	Computing Role Projection Justifications	97
6.3	Computing Subsumption Projection Justifications	98
6.3.1	Computing Subsumer Projection Justifications	100
6.3.2	Computing Subsumee Projection Justifications	102
6.4	Proofs	109
6.4.1	Proof Regarding Subsumption Projection Justification	109
6.4.2	Canonical Model & Helpful Lemmas	110
6.4.3	Proof Regarding Subsumer Projection Justification	113
6.4.4	Proof Regarding Subsumee Projection Justification	121
6.5	Evaluation	130
7	Conclusion and Future Work	131
7.1	Conclusion	131
7.2	Future Work	132
	Bibliography	135

Chapter 1

Introduction

Artificial intelligence (AI), also known as machine intelligence, has received considerable attention in recent years. The current most popular areas include deep learning, speech recognition, natural language processing. Knowledge representation, as a traditional AI area, is still a core part of AI research since it was founded in 1956 as an academic discipline. Knowledge is represented in machine processable form and logic is used for automated reasoning. Description logics in particular appear to offer a good compromise between expressive power and computational complexity of reasoning. In computer science, ontologies are mainly used to represent terminological knowledge. An ontology consists of a TBox that defines the terminology of the application domain and an ABox that states facts about a specific world. Different formats for modelling information have been recommended by the World Wide Web Consortium (W3C), most notably the Resource Description Framework (RDF) and its extensions RDF Schema (RDFS) and the Web Ontology Language (OWL). In this thesis, we consider ontologies represented using description logics. Description logics are a family of knowledge representation formalisms and form the logical underpinning of OWL. As a mature field of research, description logics offer a wealth of theoretical results on expressivity and computational complexity as well as existing implementations of reasoning systems. Applications of description logics encompass several areas including ontology engineering, web-based information systems and database management, among others. Furthermore, in this thesis, we focus on methods for extracting knowledge from ontologies that are represented using description logics for human users. To satisfy different user requirements, we propose and investigate three different approaches to knowledge extraction: ontology modules, ontology excerpts and projection modules. A module of an ontology is a subset of the ontology that captures the knowledge about the terms of interest as specified by the user. Depending on the module notion, a module may have further desired properties and provide certain logical guarantees. A drawback of modules, however, is their

relatively large size. Modules may be as large as the original ontology in the worst case. Once the ontology and the terms of interest have been fixed, there is no way to control the module size other than by changing to different module notions altogether. The notion of ontology excerpts is designed to remedy this issue. An ontology excerpt is a size-bounded subset of an ontology that preserves the knowledge that is most relevant to users' concerns.

However, due to the size restriction, ontology excerpts might be incomplete. Different to the notions of an ontology module and an ontology excerpt, projection modules preserve the answers to subsumption/instance/conjunctive queries as they follow from a reference ontology. Lastly, if users already know what they want and can express their interests as an ontology, they can use this ontology as a reference ontology and extract projection modules from the target ontology.

1.1 Ontology Modularity

Ontology modularity has been investigated for over 10 years. This area has received considerable attention resulting in a formidable literature. A challenge that remains to date is the development of practical algorithms for computing modules that are minimal w.r.t. set inclusion. The difficulty is due to the inherently high computational complexity of this task. Depending on the underlying module notion, there may be exponentially many modules that are minimal.

A module is a subset of an ontology that can act as a substitute for the ontology in certain contexts. A basic requirement for modules is to be indistinguishable from the original ontology w.r.t. a relation of inseparability. We call such *basic* modules also 'plain' modules. Further module properties such as *self-containment* and *depletion* (also called *weak* and *strong* in [25]) have been proposed in the literature [31, 27] that impose additional conditions on modules (regarding the knowledge that is contained in the module and that is left in the ontology without the module). These properties together with a range of different relations of inseparability give rise to a family of module notions. Several inseparability notions have been considered, e.g., model theoretic inseparability w.r.t. a signature [27], or inseparability w.r.t. answers to queries [35]. Popular query types are subsumption, instance and conjunctive queries. We call modules based on model theoretic inseparability *semantic modules*. This is a strong inseparability notion as in the case of \mathcal{ELH}^r -TBoxes, model theoretic inseparability w.r.t. a signature Σ coincides with entailment of second-order logic sentences over Σ (cf. Theorem 4 in [27]). In this thesis, however, we consider a weaker relation of inseparability that is based on subsumption queries between \mathcal{EL} -concepts over a given signature for an \mathcal{ELH}^r terminology. We call the resulting modules \mathcal{ELH}^r -*subsumption modules*.

An important requirement for modules is that they should be as small as possible, which is useful not only in the ontology re-use scenario [16] but also for other tasks including ontology classification, debugging, decomposition, matching and summarisation. As the smallest modules are not necessarily unique, we are interested in computing all basic \mathcal{ELH}^r -subsumption modules that are minimal w.r.t. set inclusion. Computing minimal basic semantic modules of \mathcal{ELH}^r -terminologies that are additionally self-contained and depleting has been investigated in [25, 27]. Algorithms for computing minimal modules of DL-Lite ontologies have been studied in [31]. However, to the best of our knowledge, no practical approach for computing one or all basic \mathcal{ELH}^r -subsumption modules of \mathcal{ELH}^r -terminologies has been developed so far.

Minimal modules can serve as explanations of the entire set of entailments over a signature, similar to the justifications for one consequence (i.e., minimal sets of axioms sufficient to entail the consequence) [20]. In this sense, minimal modules can improve our understanding of the internal structure of large and complex ontologies. Moreover, being able to compute all minimal modules allows us to select the smallest minimal module.

In general, extracting minimal modules is intractable, which is the reason why efficiently extractable approximations of the (union of all) minimal modules have been introduced. Among such approximations is the family of syntactic locality-based modules [16]. Such modules may contain more axioms than necessary to ensure the preservation of entailments over a signature. For instance, the size of the syntactic $\perp\top^*$ -locality modules [42] of Snomed CT,¹ the Systematized Nomenclature of Medicine – Clinical Terms, (Version Jan 2016) for 100 signatures consisting of 50 concept names selected at random together with all role names, ranges from 1 075 to 2 456 axioms. This is in contrast to the size of the minimal basic subsumption modules for these signatures that ranges from around 50 to 118 axioms. Hence, such minimal modules of Snomed CT may be more than 20 times smaller than the corresponding syntactic $\perp\top^*$ -locality modules. Implementations for extracting locality-based modules are incorporated in the OWLAPI.²

The system MEX³ has been introduced to compute minimal depleting semantic modules (which are unique for a given signature) from acyclic \mathcal{EL} -terminologies (possibly extended with inverse roles) such as Snomed CT [25]. The MEX-modules contain all minimal basic subsumption modules, which are generally smaller. The size of the MEX-modules of Snomed CT for the same signatures as above ranges from 401 to 720 axioms. However, the corresponding minimal basic subsumption modules are still at least 6 times smaller. Moreover, MEX cannot handle cyclic \mathcal{EL} -terminologies such as some recent versions of

¹<https://www.snomed.org>

²<https://github.com/owlcs/owlapi>

³<https://cgi.csc.liv.ac.uk/~konev/software/>

the National Cancer Institute’s Thesaurus (NCI).⁴ For instance, the size of the syntactic $\perp\top^*$ -locality modules of NCI (Version 14.01d) for 100 random signatures selected from NCI (as before for Snomed CT just with 100 concept names) ranges from 679 to 3 895 axioms, whereas the size of the corresponding minimal basic \mathcal{EL} -subsumption modules ranges from around 0 to 64 axioms. Clearly, the ratio of the size of the syntactic $\perp\top^*$ -locality based modules to that of the minimal basic subsumption modules is even more than 20 in this case. Another approach for extracting minimal depleting modules from DL-Lite ontologies is based on using QBF-solvers [31].

1.2 Ontology Excerpts

Knowledge about a complex system represented in ontologies yields a collection of axioms that is too large for human users to browse, let alone to comprehend or reason with. In Chapter 5 of this thesis, we propose a computational framework to zoom in on large ontologies by providing users with either the necessary axioms that act as explanations for sets of entailments, or fix-sized sub-ontologies containing the most relevant information over a vocabulary, that is signature, denoted by Σ .

Various approaches to extracting knowledge from ontologies have been suggested including ontology summarization [43, 51, 47], ontology modularization [16, 50, 48, 49, 27], ontology decomposition [11, 36], and consequence justifications [21]. Existing ontology summarization systems focus on producing an abridged version of RDF/S ontologies by identifying the most important nodes and their links under certain numeric measures, e.g., in/out degree centrality of a node [47]. Ontology modularization is also very helpful to extract knowledge from ontologies. However, computing minimal modules is known to be hard. Hence, existing systems are either restricted to tractable DLs [29, 25, 10] or they compute approximations of minimal modules [16, 11, 40].

However, different module notions and justifications share the property that the number of the axioms they contain is not bounded (besides the size of the entire ontology). Even minimal modules for small signatures may be large, rendering human understanding more difficult, which can be shown by the following experiment.

Figure 1.1 illustrates the dependency between signature size and module size in the case of SNOMED CT (Version Jan2014) consisting of 297090 axioms, 297079 concept names, and 62 role names. The coordinates of a point in Figure 1.1 are a pair (n, m) of numbers, where n corresponds to the number of terms in a signature Σ and m to the number of axioms in a subset \mathcal{S} of the ontology \mathcal{O} . The curve connects over 30 data points, each

⁴<https://ncit.nci.nih.gov/>

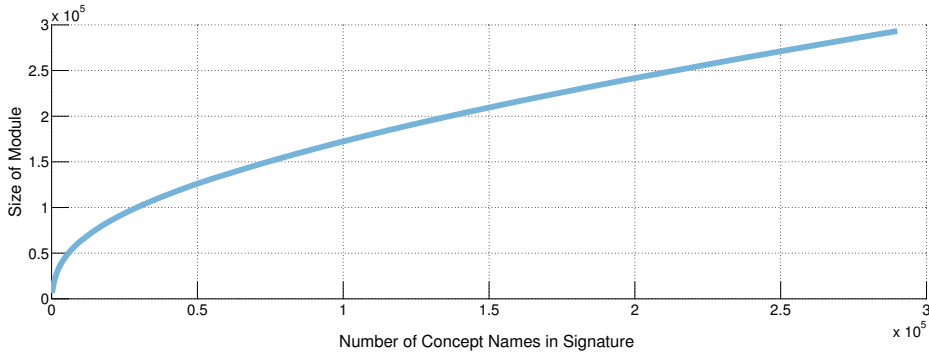


Figure 1.1 Size of $\perp\top^*$ -local module w.r.t. size of signature Σ of SNOMED CT

of which represents the median value of the sizes of 500 $\perp\top^*$ -local modules of SNOMED CT. Each module is extracted w.r.t. a signature consisting of n concept names, for n with $200 \leq n \leq 290\,000$, and 30 role names that are randomly selected from the signature of SNOMED CT. The special role name ‘RoleGroup’ is always selected. Note that we fixed the number of role names arbitrarily; similar results can be expected for different numbers of role names. The time needed to extract 500 modules ranges from about 30 min for small signatures (containing 200 concept names) to about 90 min for large signatures (containing 250\,000 concept names).⁵

We can see from Figure 1.1 that the module sizes increase with the size of the input signature. For small signatures, the slope is steep, showing that the modules are relatively large compared to the signature size. However, with increasing signature sizes, the slope flattens. For different signatures of the same size, there are still variations in the sizes of the modules for these signatures. In this experiment, the module sizes vary from 2\,633 to 4\,086 for signatures up to 100\,000 concept names and 30 role names. For larger signatures the variation in module size reduces to 224 for signatures with 290\,000 concept names and 30 role names, and converges to 0 as signature is expanded to the whole signature of SNOMED CT.

Let us consider the coordinates of a point in the chart in Figure 1.1 as a pair (n, m) of numbers, where n corresponds to the size of a signature Σ and m to the size of a subset \mathcal{S} of the ontology \mathcal{O} . Let $\mathcal{M}_\Sigma(\mathcal{O})$ be the module of the ontology \mathcal{O} w.r.t. the signature Σ (under any module notion). Note that, for a signature Σ' and a subset $\mathcal{S}' \subseteq \mathcal{O}$ that correspond to a point in the area above the curve for the ontology \mathcal{O} in Figure 1.1, we may have that $\mathcal{M}_{\Sigma'}(\mathcal{O}) \not\subseteq \mathcal{S}'$. In this case, \mathcal{S}' likely contains axioms that do not contribute to the meaning of the symbols in Σ' . Therefore, we are mainly interested in the area below the curve for an ontology \mathcal{O} . Let (n, m) be a point in that area, and let $\mathcal{S} \subseteq \mathcal{O}$ and Σ be such that $|\Sigma| = n$ and

⁵The experiments were conducted on a PC equipped with an Intel Xeon E5-2640 CPU running at 2.50GHz and with 100GB of RAM. We used Debian GNU/Linux 7.3 as operating system, Java version 1.7.0_51 and OWLAPI version 3.4.8.

$|\mathcal{S}| = m$. We have that \mathcal{S} contains fewer axioms than the module $\mathcal{M}_\Sigma(\mathcal{O})$ (not considering the variation of module sizes), i.e., $|\mathcal{S}| \leq |\mathcal{M}_\Sigma(\mathcal{O})|$. Therefore, \mathcal{S} is likely to be incomplete in capturing the meaning of the symbols in Σ . The trade-off for obtaining full control over the size of \mathcal{S} is a certain degree of incompleteness of \mathcal{S} . To this end, the notion of best excerpts is inspired as size-bounded subsets of ontologies that preserve as much knowledge about a given signature as possible.

The following real-world example illustrates possible benefits of best excerpts. The example axioms come from Snomed CT (FS for Finding_site, RG for Role_Group).

$$\alpha_1 := \text{Decreased_blood_volume} \sqsubseteq \text{Cardiovascular_finding}$$

$$\alpha_2 := \text{Cardiovascular_shunt} \sqsubseteq \text{Cardiovascular_finding}$$

$$\alpha_3 := \text{Cardiac_shunt} \sqsubseteq \text{Cardiovascular_shunt} \sqcap \exists \text{RG} . (\exists \text{FS} . \text{Heart_structure})$$

$$\alpha_4 := \text{Cardiovascular_structure} \sqsubseteq \text{Body_system_structure}$$

Suppose a user is concerned with the cardiovascular disease defined in the Snomed CT ontology \mathcal{T} consisting of around 300000 axioms. The user then selects the terms `Cardiovascular_finding`, `Decreased_blood_volume` and `Cardiac_shunt` from \mathcal{T} as her signature Σ of interest. To help the user zoom in on \mathcal{T} for Σ , we can extract, for instance, the $\perp\top^*$ -module and obtain 51 axioms, or the *smallest* minimal modules, which yields a further reduction down to 15 axioms, among which are the axioms given above (α_1 - α_4). Arguably our user still feels overwhelmed by the amount of 15 axioms. This is where the notion of best k -excerpt steps in. By setting $k = 3$, the user can get a best 3-excerpt \mathcal{E}_1 consisting of the axioms 1–3 listed above. By zooming in further, say extracting one-sized excerpts, she obtains \mathcal{E}_2 consisting of the first axiom. As a best excerpt, \mathcal{E}_1 guarantees all logical entailments over the terms `Cardiac_shunt` and `Decreased_blood_volume`. And the singleton \mathcal{E}_2 keeps the complete information over the term `Decreased_blood_volume`. Note that \mathcal{E}_2 is returned due to the fact that it needs more than two axioms to preserve the full information for any other concept in Σ . Moreover, axiom 4 is missing in \mathcal{E}_1 and in \mathcal{E}_2 . This is because the latter merely serve to provide background knowledge for reasoning over, thus not directly linked to, the user's input terms Σ , which are excluded from best excerpts due to the size restriction. In this way, the user is able to desired size that captures the most relevant information for her signature of interest, thus gains size control over a large ontology.

1.3 Projection Modules

Consider the scenario of the medical condition multiple myeloma, also known as bone marrow cancer (BMCancer). This is a cancer of plasma cells, an antibody producing type of white blood cell. The affected blood cells produce the protein albumin that can damage the kidneys. Damaged kidneys

may be detected in a urine test (Urinalysis) as the presence of increased amounts of albumin in the urine. Normal protein concentrations in urine should be low. Depending on the amounts of albumin, this finding is classified as microalbuminuria (MA). The cancerous cells activate certain bone cells (osteoclasts) to resorb bone tissue causing the release of calcium into the blood (leading to a possible Hypercalcemia). The disruption of red blood cell production in the bone marrow may cause anemia, and the breakdown of bone may eventually result in a broken back (Spinal_Fracture).

Suppose a medical terminology \mathcal{T}_1 consisting of axioms $\alpha_1, \dots, \alpha_6$ generally states that spinal fractures should be treated with orthopedic surgery and classifies multiple myeloma as bone marrow cancer. It also states that patients with multiple myeloma should be given cancer treatment. We can think of the information in \mathcal{T}_1 as a basic classification and guideline that should be complemented with more specific domain knowledge on symptoms, diagnostic procedures and types of treatments where applicable.

We further suppose that domain experts together with ontology engineers in fact have, based on \mathcal{T}_1 , implemented a more specific terminology \mathcal{T}_2 including the axioms $\alpha_2, \dots, \alpha_6$ and β_1, \dots, β_5 , among many others. In the development of \mathcal{T}_2 , the axiom α_1 has been extended to β_1 to include more information on prescribed treatments in the case of a spinal fracture. In particular, β_1 additionally states that a urinalysis should be performed to check for a possibly increased concentration of the protein albumin as well as medical imaging to detect any bone lesions. Each of these findings can be essential in diagnosing multiple myeloma, which may otherwise be undetected and, thus, have fatal consequences. The other axioms from \mathcal{T}_1 have been preserved in \mathcal{T}_2 , whereas β_2, \dots, β_5 are new axioms classifying treatments and describing symptoms of multiple myeloma.

$$\alpha_1 := \text{Spinal_Fracture} \sqsubseteq \text{Fracture}$$

$$\alpha_2 := \text{Fracture} \sqsubseteq \exists \text{prescribe.Orthopedic_Surgery}$$

$$\alpha_3 := \text{Orthopedic_Surgery} \sqsubseteq \text{Treatment}$$

$$\alpha_4 := \text{Multiple_Myeloma} \sqsubseteq \text{BMCancer}$$

$$\alpha_5 := \text{BMCancer} \sqsubseteq \exists \text{prescribe.Cancer_Treatment}$$

$$\alpha_6 := \text{Cancer_Treatment} \sqsubseteq \text{Treatment}$$

$$\beta_1 := \text{Spinal_Fracture} \sqsubseteq \text{Fracture} \sqcap \exists \text{prescribe.Urinalysis} \sqcap \exists \text{prescribe.Medical_Imaging}$$

$$\beta_2 := \text{Multiple_Myeloma} \sqsubseteq \text{MA} \sqcap \text{Anemia} \sqcap \text{Hypercalcemia}$$

$$\beta_3 := \text{MA} \equiv \exists \text{prescribe.}(\text{Urinalysis} \sqcap \exists \text{has_finding.Increased_Albumin})$$

$$\beta_4 := \text{Medical_Imaging} \sqsubseteq \text{Treatment}$$

$$\beta_5 := \text{Urinalysis} \sqsubseteq \text{Treatment}$$

To promote quality and safety in the health care system, the integrity of \mathcal{T}_2 needs to be verified. This means checking that \mathcal{T}_2 correctly implements the knowledge regarding the terms, say, `Spinal_Fracture`, `Treatment`, `Multiple_Myeloma`, `prescribe`, and `has_finding` as used in \mathcal{T}_1 (underlined). It also means checking that \mathcal{T}_2 satisfies certain restrictions regarding the formulation of axioms as well as term usage. In an effort to ensure integrity of \mathcal{T}_2 w.r.t. \mathcal{T}_1 and compliance of \mathcal{T}_2 with guidelines, the axioms from \mathcal{T}_2 that capture \mathcal{T}_1 's knowledge need to be inspected. Guidelines by health insurances may require hospitals to avoid unnecessary treatments so as to avoid medical waste. Additionally guidelines regarding the style or form of axioms may apply such as restrictions on term usage or the requirement to use certain design patterns.

A module of \mathcal{T}_2 for the terms, however, preserves all knowledge of \mathcal{T}_2 regarding these terms. The resulting axioms may be too numerous for it to be feasible to inspect them manually. In our illustrating example, a comprehensive module notion would, when applied to \mathcal{T}_2 w.r.t. the terms, yield all axioms of \mathcal{T}_2 as shown above. A module notion that is minimal (w.r.t. set inclusion) would yield the axioms β_1, \dots, β_5 . These axioms preserve \mathcal{T}_1 's entailments `Spinal_Fracture` \sqsubseteq `\exists prescribe.Treatment` and `Multiple_Myeloma` \sqsubseteq `\exists prescribe.Treatment`, but they also entail, e.g.,

$$\gamma := \text{Multiple_Myeloma} \sqsubseteq \exists \text{prescribe.has_finding}.\top,$$

which does not follow from \mathcal{T}_1 . On the other hand, the following subsets of \mathcal{T}_2 that do preserve the entailments of \mathcal{T}_1 are not minimal modules of \mathcal{T}_2 as they do not entail γ : $\{\alpha_2, \dots, \alpha_6, \beta_1\}$, $\{\alpha_4, \alpha_5, \alpha_6, \beta_1, \beta_4\}$, and $\{\alpha_4, \alpha_5, \alpha_6, \beta_1, \beta_5\}$. Each of these subsets needs to be inspected to ensure integrity and compliance of all relevant parts of \mathcal{T}_2 . For the inspection effort to be feasible, every such subset needs to be *as small as possible*.

To sum up, the task is to assist users to check how \mathcal{T}_1 's knowledge is implemented in \mathcal{T}_2 . The solution is to extract only those subsets from \mathcal{T}_2 that preserve the mere knowledge of \mathcal{T}_1 and that are minimal w.r.t. set inclusion so as to enable processing by a human, which cannot be produced by existing module notions.

Best practices in engineering individual or networked systems of ontologies include the use of design patterns [37] and requirement specifications [45]. Design patterns are syntactic templates of axioms that are to be instantiated with terms. Requirements specifications help by facilitating the ontology development process including the application of design patterns as well as verification of the resulting ontologies.

Various approaches to comparing ontologies have been suggested, including ontology mapping, or ontology alignment [13], and logical difference [28, 30, 34]. Ontology matching is the process of determining correspondences, in particular the subsumption, equivalence, or disjointness relation between two concept or relation names from different ontologies. A good concept similarity [33, 1] is often helpful for ontology matching. In contrast, logical difference focuses on the comparison of entailed logical consequences from each ontology and returns difference witnesses if differences are present.

For latest results on inseparability (in particular inseparability w.r.t. conjunctive queries), see, e.g., [7, 8, 19], and for a survey on query inseparability, see, e.g., [6].

To compute projection modules, in this thesis, we generalize the notion of justification to *subsumption justification* as a minimal set of axioms that maintains a consequence. Currently, the approaches for computing all the justifications of an ontology w.r.t. a consequence can be classified into two categories: “glass-box” [5, 21, 2, 22] and “black-box” [21, 52, 46]. “Black-box” algorithms use existing reasoners to check whether a concept inclusion follows on from an ontology while searching for possible sub-ontology that can serve as justifications. Differently, “glass-box” approaches modify a pre-existing reasoner implementation in order to keep track of which axioms were used during the classification of an ontology. Since it is difficult to avoid generating duplicate inferences for the same consequence and there might exist exponentially many justifications for a given ontology and a concept inclusion in the worst case, computing all justifications in general is a difficult task. It is however possible to compute one justification in polynomial time.

1.4 Structure of the Thesis

The structure of the thesis is listed as follows:

Chapter 2 In this chapter, we introduce the relevant notions that are necessary for this thesis. We start by the foundations of Description Logics \mathcal{EL} and then we review the theory of logical differences, relevant lemmas and theorems. In addition, some existing ontology module notions are recalled. Finally, we introduce the notion of ontology justification and partial Max-SAT problem.

Chapter 3 We introduce the notion of *subsumption justification* to capture the subsumption knowledge concerning a term with respect to all primitive and complex concepts built from terms in a given vocabulary Σ . It extends the notion of classical justification that is a minimal set of axioms needed to preserve the entailment of a particular subsumption $C \sqsubseteq D$.

We provide two dedicated simulation notions to characterise the set of subsumers and the set of subsumees formulated over a target signature Σ for a given signature term X w.r.t. an \mathcal{ELH} - and \mathcal{ELH}^r -terminology \mathcal{T} . The simulation notions originate from the proof-theoretic approach from [34] developed for the problem of deciding the logical difference between ontologies [23]. Based on the simulation notions, we devise recursive algorithms for extracting the minimal subsets of axioms that preserve the entailments of all Σ -subsumers and all Σ -subsumees of X w.r.t. \mathcal{T} . We show that the respective subsumer and subsumee justifications obtained in this way can then be combined to yield subsumption justifications.

Finally, we evaluate a prototype implementation for computing subsumption justifications over large biomedical terminologies. The results are encouraging as they show that computing subsumption justifications is indeed feasible in several important practical cases.

Chapter 4 In this chapter, we introduce a module notion, called basic \mathcal{ELH}^r -subsumption modules and present two different approaches to compute them. In section 4.2, we extend the black-box approach for finding one or all justifications in [20], which is based on Reiter’s hitting set algorithm [38]. Instead of ensuring that a given entailment is preserved, we introduce an oracle to determine the inseparability between the original ontology and the resulting module. As an oracle we use a variant of the system CEX, which is a tool for deciding whether two \mathcal{ELH}^r -terminologies are logically different w.r.t. a signature [28, 24, 34]. Additionally, several optimisations to speed up the computation of minimal modules are investigated.

We also present an experimental evaluation of our algorithms by applying them on the prominent and large medical ontologies Snomed CT and NCI. We note that our algorithms are applicable to ontologies formulated in any ontology language provided that a tool is available that can effectively decide the inseparability relation. As CEX works with variants of \mathcal{ELH}^r -terminologies, we restrict the presentation of our algorithms to \mathcal{ELH}^r -terminologies.

An alternative to the black-box approach for computing minimal modules is to directly select the relevant axioms from an ontology that preserve the desired entailments. Such a *glass-box approach* has been investigated for acyclic \mathcal{ELH} -terminologies (i.e., without domain and range restrictions of roles) in Section 4.3. This technique is based on subsumption justification that we introduced in Chapter 3. In addition, in this thesis we introduce a technique for computing even smaller (but possibly incomplete) modules called *ontology excerpts*.

Chapter 5 A best k -excerpt of a TBox is a sub-ontology that contains at most k axioms that best capture the knowledge about users’ interests in terms of weighted signature. In Chapter 5, we encode the problem of selecting the right subsumption modules w.r.t. a signature as a partial Max-SAT problem based on subsumption justifications that we introduced in Chapter 3. Overall we obtain an algorithm for computing best k -excerpts w.r.t. a signature. Then we introduce weighted signature and update the encoding method as a weighted partial Max-SAT problem to computing best excerpts for weighted signature. We also propose a method to rank axioms in excerpts when presenting users a best excerpt. Finally, we evaluate our algorithm by using it with the prominent bio-medical ontologies Snomed CT. We demonstrate that computing best k -excerpts is a viable approach for the task of summarizing large ontologies with a controllable small number of axioms that are most relevant for a given set of terms.

Chapter 6 We introduce a novel module notion called *projection module* that entails the queries that follow on from a reference ontology. We consider subsumption, instance and conjunctive queries to represent the knowledge to be preserved. We develop algorithms for computing all minimal projection modules of \mathcal{ELH}^r -terminologies, inspired by the simulation notions developed for deciding the logical difference problem [34]. Our preliminary evaluation on real world biomedical ontologies shows that the projection module notion creates a new dimension to understand relations among multiple ontologies.

Chapter 7 In Chapter 7 we summarize the results and indicate some possible future work based on this thesis.

Chapter 2

Preliminaries

In this chapter we review the theoretical foundation that the following chapters are based on. We start by explaining the basic notions of the Description Logic \mathcal{EL} and some of its extensions. We focus on these light-weight Description Logics as we deal in this thesis with the knowledge that is represented in such formalisms. Then we review the theory of logical difference, which plays an important role in the remainder of this thesis. In Section 2.3, we have a close look at existing ontology module notions. We follow by introducing the notion of justification, which we will use in Chapter 3. Finally, we introduce the foundation of the partial Max-SAT problem, which will be used in Chapter 5.

2.1 Description Logic \mathcal{EL} and its Extensions

In this section, we are introducing the family of Description Logics \mathcal{EL} . We start by briefly reviewing the Description Logic \mathcal{EL} and several of its extensions with range restrictions and conjunction of roles and the universal role as well as concept subsumptions based on these extensions. For a more detailed introduction to description logics, we refer to [3].

In the remainder of the thesis, we denote the sets of concept and role names by N_C and N_R , respectively. We assume these sets to be mutually disjoint and countably infinite. Capital letters such as A, B, X, Y, Z will denote concept names from N_C , whereas lower-case letters r, s, t will denote role names from N_R .

Definition 1 (Syntax of \mathcal{EL} -Family Concepts) *The sets of \mathcal{EL} -concepts C , $\mathcal{EL}^{\text{ran}}$ -concepts D , \mathcal{EL}^{\square} -concepts E , and $\mathcal{EL}^{\square, u}$ -concepts F are built according to the grammar rules:*

$$\begin{aligned} C &::= \top \mid A \mid C \sqcap C \mid \exists r.C \mid \text{dom}(r) \\ D &::= \top \mid A \mid D \sqcap D \mid \exists r.D \mid \text{dom}(r) \mid \text{ran}(r) \\ E &::= \top \mid A \mid E \sqcap E \mid \exists R.E \\ F &::= \top \mid A \mid F \sqcap F \mid \exists R.F \mid \exists u.F \end{aligned}$$

where $A \in \mathbf{N}_C$, $r \in \mathbf{N}_R$, u is a fresh logical symbol (the universal role) and $R = r_1 \sqcap \dots \sqcap r_n$ with $r_1, \dots, r_n \in \mathbf{N}_R$, for $n \geq 1$.

Definition 2 (Semantics of \mathcal{EL} Concepts) An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of a non-empty set, the domain $\Delta^{\mathcal{I}}$, and an interpretation function $\cdot^{\mathcal{I}}$ that maps each concept name A to a subset $A^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$, every role name r to a binary relation $r^{\mathcal{I}}$ over $\Delta^{\mathcal{I}}$ and the universal role u to $u^{\mathcal{I}} = \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. The extension $C^{\mathcal{I}}$ of an \mathcal{EL} -concept C is defined inductively as follows:

- $\top^{\mathcal{I}} := \Delta^{\mathcal{I}}$,
- $(C \sqcap D)^{\mathcal{I}} := C^{\mathcal{I}} \cap D^{\mathcal{I}}$,
- $(\exists r.C)^{\mathcal{I}} := \{x \in \Delta^{\mathcal{I}} \mid \exists y \in C^{\mathcal{I}} : (x, y) \in r^{\mathcal{I}}\}$,
- $(\text{dom}(r))^{\mathcal{I}} := \{x \in \Delta^{\mathcal{I}} \mid \exists y : (x, y) \in r^{\mathcal{I}}\}$,
- $(\text{ran}(r))^{\mathcal{I}} := \{y \in \Delta^{\mathcal{I}} \mid \exists x : (x, y) \in r^{\mathcal{I}}\}$ and
- $(r_1 \sqcap \dots \sqcap r_n)^{\mathcal{I}} := r_1^{\mathcal{I}} \cap \dots \cap r_n^{\mathcal{I}}$.

Example 3 To illustrate concept descriptions, consider an example from the biomedical domain. We assume that *Finding_or_region_of_thorax* and *Mediastinal_Structure* are concept names, and *finding_site* is a role name. We can specify the concept of *Finding_or_region_of_thorax* that can be found in the *Mediastinal_Structure* by using the following concept:

$$\text{Finding_or_region_of_thorax} \sqcap \exists \text{finding_site. Mediastinal_Structure.}$$

Definition 4 (Concept and Role Inclusions, and TBoxes) Let C be an \mathcal{EL} -concept, D an $\mathcal{EL}^{\text{ran}}$ -concept, F an $\mathcal{EL}^{\sqcap, u}$ -concept and $r, s \in \mathbf{N}_R$. The sets of \mathcal{EL} \mathcal{H} -inclusions α , $\mathcal{EL}^{\mathcal{H}^r}$ -inclusions β , $\mathcal{EL}^{\text{ran}}$ -inclusions γ and $\mathcal{EL}^{\text{ran}, \sqcap, u}$ -inclusions δ are built according to the grammar rules:

$$\begin{aligned} \alpha &::= C \sqsubseteq C \mid C \equiv C \mid r \sqsubseteq s \\ \beta &::= C \sqsubseteq C \mid \text{ran}(r) \sqsubseteq C \mid \text{ran}(r) \sqcap C \sqsubseteq C \mid C \equiv C \mid r \sqsubseteq s \\ \gamma &::= D \sqsubseteq C \mid r \sqsubseteq s \\ \delta &::= D \sqsubseteq F \mid r \sqsubseteq s \end{aligned}$$

An \mathcal{L} -TBox is a finite set of \mathcal{L} -inclusions, where \mathcal{L} ranges over the set of \mathcal{EL} \mathcal{H} , $\mathcal{EL}^{\mathcal{H}^r}$, $\mathcal{EL}^{\text{ran}}$ and $\mathcal{EL}^{\text{ran}, \sqcap, u}$.

We refer to inclusions also as *axioms*.

Definition 5 (Instance Assertions and ABoxes) Let \mathbf{N}_I be a set of individuals names and let $a, b \in \mathbf{N}_I$. Additionally, let C be an \mathcal{L} -concept and $r \in \mathbf{N}_R$, where \mathcal{L} ranges over the set of \mathcal{EL} , $\mathcal{EL}^{\text{ran}}$, \mathcal{EL}^{\sqcap} , and $\mathcal{EL}^{\sqcap, u}$. A \mathcal{L} -instance assertion is an expression of the form:

$$C(a) \quad \text{or} \quad r(a, b).$$

An \mathcal{L} -ABox is a finite set of \mathcal{EL} -instance assertions.

Definition 6 (Semantics of Concepts, Inclusions, Assertions, TBoxes and ABoxes) An interpretation \mathcal{I} satisfies a concept C , an axiom $C \sqsubseteq D$, $C \equiv D$ or $r \sqsubseteq s$, and an \mathcal{EL} -instance assertion $C(a)$ or $r(a, b)$ iff, respectively:

- $C^{\mathcal{I}} \neq \emptyset$,
- $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$,
- $C^{\mathcal{I}} = D^{\mathcal{I}}$,
- $r^{\mathcal{I}} \subseteq s^{\mathcal{I}}$,
- $a^{\mathcal{I}} \in C^{\mathcal{I}}$, or
- $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$.

We write $\mathcal{I} \models \varphi$ iff the interpretation \mathcal{I} satisfies the axiom or instance assertion φ . An interpretation \mathcal{I} satisfies a TBox \mathcal{T} (or an ABox \mathcal{A}), written $\mathcal{I} \models \mathcal{T}$ ($\mathcal{I} \models \mathcal{A}$), iff for every axiom $\varphi \in \mathcal{T}$, it holds that $\mathcal{I} \models \varphi$ (for every instance assertion $\lambda \in \mathcal{A}$, $\mathcal{I} \models \lambda$). A TBox \mathcal{T} entails an axiom φ or, equivalently, φ follows from \mathcal{T} , written $\mathcal{T} \models \varphi$, iff for all interpretations \mathcal{I} , $\mathcal{I} \models \mathcal{T}$ implies $\mathcal{I} \models \varphi$. A TBox \mathcal{T} and an ABox \mathcal{A} together entail an instance assertion λ or, equivalently, λ follows from \mathcal{T} and \mathcal{A} , written $(\mathcal{T}, \mathcal{A}) \models \lambda$, iff for all interpretations \mathcal{I} , $\mathcal{I} \models \mathcal{T}$ and $\mathcal{I} \models \mathcal{A}$ implies $\mathcal{I} \models \lambda$.

Note that each of the concepts introduced above is satisfiable.

Example 7 Considering the following concept and role inclusions in the biomedical domain:

$$\begin{aligned} \alpha_1 &::= \text{Color_finding}(\text{finding}) \equiv \text{Clinical_finding}(\text{finding}) \sqcap \exists \text{interprets}(\text{attribute}).\text{Color} \\ \alpha_2 &::= \text{Disease}(\text{disorder}) \sqsubseteq \text{Clinical_finding}(\text{finding}) \\ \alpha_3 &::= \text{interprets}(\text{attribute}) \sqsubseteq \text{concept_module_object_attribute}(\text{attribute}) \end{aligned}$$

Axiom α_1 defines the concept name $\text{Color_finding}(\text{finding})$ in terms of the concept names $\text{Clinical_finding}(\text{finding})$ and Color as well as the role name $\text{interprets}(\text{attribute})$ by stating that color findings are exactly the clinical findings that interpret colors. Axiom α_2 specifies the concept name $\text{Disease}(\text{disorder})$ in terms of $\text{Clinical_finding}(\text{finding})$ by stating that a disease is a clinical finding. Finally, Axiom α_3 defines $\text{interprets}(\text{attribute})$ in terms of the role name $\text{concept_module_object_attribute}(\text{attribute})$ by stating that every relationship among the property ‘interprets’ also belongs to the property ‘concept module object attribute’.

Definition 8 (\mathcal{ELH} -terminology) An \mathcal{ELH} -terminology \mathcal{T} is an \mathcal{ELH} -TBox consisting of axioms α of the following form:

- $A \sqsubseteq C$,
- $A \equiv C$,
- $r \sqsubseteq s$, or
- $\text{dom}(r) \sqsubseteq C$,

where $A \in \mathbf{N}_C$, $r, s \in \mathbf{N}_R$, C is an \mathcal{EL} -concept and no concept name occurs more than once on the left-hand side of an axiom.¹

Compared to \mathcal{ELH} -terminologies, an \mathcal{ELH}^r -terminology additionally allows for range restrictions.

Definition 9 (\mathcal{ELH}^r -terminology) An \mathcal{ELH}^r -terminology \mathcal{T} is an \mathcal{ELH}^r -TBox consisting of axioms α of the following form:

- $A \sqsubseteq C$,
- $A \equiv C$,
- $r \sqsubseteq s$,
- $\text{dom}(r) \sqsubseteq C$ or
- $\text{ran}(r) \sqsubseteq C$,

where $A \in \mathbf{N}_C$, $r, s \in \mathbf{N}_R$, C is an \mathcal{EL} -concept and no concept name occurs more than once on the left-hand side of an axiom.

Example 10 (Ex. 7 contd.) We complement axioms α_1 – α_3 with four more axioms α_4 – α_7 from the biomedical domain:

$\alpha_1 := \text{Color_finding}(\text{finding}) \equiv \text{Clinical_finding}(\text{finding}) \sqcap \exists \text{interprets}(\text{attribute}).\text{Color}$

$\alpha_2 := \text{Disease}(\text{disorder}) \sqsubseteq \text{Clinical_finding}(\text{finding})$

$\alpha_3 := \text{interprets}(\text{attribute}) \sqsubseteq \text{concept_module_object_attribute}(\text{attribute})$

$\alpha_4 := \text{Normal_color}(\text{finding}) \sqsubseteq \text{Color_finding}(\text{finding})$

$\alpha_5 := \text{Normal_color}(\text{finding}) \sqsubseteq \exists \text{interprets}(\text{attribute}).\text{Color}(\text{observable_entity})$

$\alpha_6 := \text{ran}(\text{finding_sites}) \sqsubseteq \text{Body_Structure}(\text{body_structure})$

$\alpha_7 := \text{Color_finding}(\text{finding}) \equiv \text{Diagnostic_result}(\text{finding}) \sqcap \exists \text{has_finding}(\text{attribute}).\text{Color}$

Axioms α_4 and α_5 both specify the concept name $\text{Normal_color}(\text{finding})$, the former in terms of the concept name $\text{Color_finding}(\text{finding})$ and the latter in terms of the role name $\text{interprets}(\text{attribute})$ and the concept name $\text{Color}(\text{observable_entity})$. Then, Axiom α_4 states that a normal color is a color

¹A concept equation $A \equiv C$ stands for the inclusions $A \sqsubseteq C$ and $C \sqsubseteq A$.

finding, and α_5 states that a normal color can be interpreted as an observable color. Next, Axiom α_6 specifies the role name `finding_sites` in terms of the concept name `Body_Structure(body_structure)` by restricting the range of the property 'finding sites' to refer to body structures. Finally, Axiom α_7 provides an alternative definition of `Color_finding(finding)` in terms of the role name `has_finding(attribute)` and the concept names `Diagnostic_result(finding)` and `Color` by stating that color findings are exactly the diagnostic results that have a finding which is a color.

We have that $\mathcal{T} = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4\}$ is an \mathcal{ELH}^r -terminology and $\mathcal{T} \cup \{\alpha_6\}$ is an \mathcal{ELH}^r -terminology. The set $\mathcal{T}' = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6\}$ is an \mathcal{ELH}^r -TBox, but, according to Definition 9, it is not an \mathcal{ELH}^r -terminology as the concept name `Normal_color(finding)` occurs on the left-hand side of two axioms, i.e. α_4 and α_5 . Let \mathcal{T}'' be a TBox obtained from \mathcal{T}' by replacing α_4 and α_5 with Axiom α_8 .

$$\alpha_8 := \text{Normal_color(finding)} \sqsubseteq \text{Color_finding(finding)} \sqcap \\ \exists \text{interprets(attribute).Color(observable_entity)}$$

\mathcal{T}'' is an \mathcal{ELH}^r -terminology, and, moreover, \mathcal{T}'' is semantically equivalent to \mathcal{T}' (i.e. all axioms from \mathcal{T}' are entailed by \mathcal{T}'' , and vice versa). Similarly, according to Definition 8, the set $\mathcal{T} \cup \{\alpha_7\}$ is not an \mathcal{ELH}^r -terminology due to the concept name `Color_finding(finding)` occurring on the left-hand sides of axioms α_1 and α_7 . In this case, however, we cannot simply replace α_1 and α_7 with a fresh axiom that combines the two.

To simplify the presentation, we assume that terminologies do not contain axioms of the form $A \equiv B$ or $A \equiv \top$ (after having removed multiple \top -conjuncts) for concept names A and B .

Definition 11 (Signature) A signature Σ is a finite set of symbols from \mathbb{N}_C and \mathbb{N}_R . We write $\text{sig}(\chi)$ for the signature of χ as the set of concept and role names occurring in χ , where χ ranges over any syntactic object. Additionally, we define that $\text{sig}^V(\chi) := \text{sig}(\chi) \cap V$, where V is a set.

The symbol Σ is used as a subscript (or a prefix) to a set of concepts or inclusions to denote that the elements only use symbols from Σ , e.g., \mathcal{EL}_Σ , $\mathcal{EL}_\Sigma^{\text{ran}}$, $\mathcal{EL}_\Sigma^\square$, $\mathcal{EL}_\Sigma^{\square, u}$, \mathcal{ELH}_Σ^r , Σ -ABox or Σ -query etc.

Definition 12 (Acyclic and Cyclic Terminology) Let \mathcal{T} be a terminology and $\prec_{\mathcal{T}}$ be the binary relation over \mathbb{N}_C defined by $A \prec_{\mathcal{T}} B$ iff there is an axiom of the form $A \sqsubseteq C$ or $A \equiv C$ in \mathcal{T} such that $B \in \text{sig}(C)$. A terminology \mathcal{T} is acyclic if the transitive closure $\prec_{\mathcal{T}}^+$ of $\prec_{\mathcal{T}}$ is irreflexive; otherwise \mathcal{T} is cyclic.

Definition 13 (Conjunctive Concept Name) A concept name A is called conjunctive in \mathcal{T} iff there exists an axiom of the form $A \equiv B_1 \sqcap \dots \sqcap B_n$, $n \geq 2$ in \mathcal{T} ; otherwise A is said to be non-conjunctive in \mathcal{T} . We denote by $B_i \in \text{non-conj}_{\mathcal{T}}(A)$ and $1 \leq i \leq n$.

Definition 14 (Normalised $\mathcal{EL}\mathcal{H}^r$ -Terminology) Let $\varphi \in \{A, \text{dom}(s), \text{ran}(s)\}$, A, B, B_i be concept names, and let r, s be role names. Additionally, let each conjunct B_i be non-conjunctive in \mathcal{T} . An $\mathcal{EL}\mathcal{H}^r$ -terminology \mathcal{T} is normalised iff \mathcal{T} only contains axioms of the following forms:

- $r \sqsubseteq s$,
- $\varphi \sqsubseteq B_1 \sqcap \dots \sqcap B_n$ with $n \geq 1$,
- $A \sqsubseteq \exists r.B$,
- $A \sqsubseteq \text{dom}(r)$,
- $A \equiv B_1 \sqcap \dots \sqcap B_m$ with $m \geq 2$,
- $A \equiv \exists r.B$.

Every $\mathcal{EL}\mathcal{H}^r$ -terminology \mathcal{T} can be normalised in polynomial time such that the resulting terminology is a conservative extension of \mathcal{T} [23].

Example 15 (Ex. 7 contd.) The axiom α_1 in Example 7 can be normalised resulting in, e.g., the following two new axioms:

- $\text{Color_finding}(\text{finding}) \equiv \text{Clinical_finding}(\text{finding}) \sqcap A$,
- $A \equiv \exists \text{interprets}(\text{attribute}).\text{Color}$,

where A is a new concept name introduced by the normalization of α_1 .

Any $\mathcal{EL}\mathcal{H}^r$ -terminology can be normalised by appropriately replacing nested complex concepts C by fresh concept names X_C and adding concept equations $X_C \equiv C$ to the terminology that define the new symbols. It can readily be seen that this transformation is tractable and that it does not change the meaning of the original TBox. The following lemma from [23] makes this precise.

Lemma 16 For every $\mathcal{EL}\mathcal{H}^r$ -terminology \mathcal{T} , there is a normalised $\mathcal{EL}\mathcal{H}^r$ -terminology \mathcal{T}' of polynomial size in the size of \mathcal{T} such that $\mathcal{T} \equiv_{\Sigma} \mathcal{T}'$ with $\Sigma = \text{sig}(\mathcal{T})$.

2.2 Logical Difference

In this section, we briefly recall basic notions related to logical difference between ontologies and some lemmas that we will use later in the thesis [23, 28].

Before we give the formal definitions of logical difference, first we recall the notions of instance and conjunctive queries. While *instance queries* are simply instance assertions (cf. Definition 5), conjunctive queries require their own definition.

Definition 17 (Conjunctive Query) Let N_I and N_V be disjoint sets of individual and variable names. A conjunctive query is a first-order formula built according to the following format:

$$\exists y_1 \dots \exists y_n. \bigwedge_{i \in I_1} A_i(s_i) \wedge \bigwedge_{j \in I_2} r_j(t_j, t'_j),$$

where $y_1, \dots, y_n \in N_V$ for $n \geq 1$ are variable names, I_1, I_2 are finite sets of indices, and for $i \in I_1$ and $j \in I_2$, A_i ranges over concept names in N_C , r_j ranges over role names in N_R , and s_i, t_j, t'_j range over individual and variable names in $N_I \cup N_V$.

The notion of entailment of an instance query from a TBox and ABox generalises to conjunctive queries.

Example 18 We illustrate instance and conjunctive queries with the following example from the medical domain. Let \mathcal{T} be a TBox and \mathcal{A} an ABox, where

$$\begin{aligned} \mathcal{T}: \quad & \text{Doctor} \equiv \text{Employee} \sqcap \exists \text{treat.Disease} \\ & \text{ran}(\text{treat}) \sqsubseteq \text{Disease} \end{aligned}$$

$$\begin{aligned} \mathcal{A}: \quad & \text{Employee}(\text{Ping}) \\ & \text{Disease}(\text{Anemia}) \\ & \text{treat}(\text{Ping}, \text{Anemia}) \end{aligned}$$

Let $\lambda = (\text{Employee} \sqcap \exists \text{treat.Disease})(\text{Ping})$ be an instance query and $q(x) = \exists y. \text{Employee}(x) \wedge \text{treats}(x, y)$ a conjunctive query. Note that $q(x)$ has one free variable, x . It can readily be verified that $(\mathcal{T}, \mathcal{A}) \models \lambda$ holds according to Definition 6. Moreover, it is not hard to see that $(\mathcal{T}, \mathcal{A}) \models q(x)$.

Definition 19 (Logical Difference) Let \mathcal{T}_1 and \mathcal{T}_2 be \mathcal{ELH}^r -TBoxes and let Σ be a signature. Additionally, let $\mathcal{L} \in \{\mathcal{ELH}^r, \mathcal{EL}^{\text{ran}}, \mathcal{EL}^{\text{ran}, \sqcap, \cup}\}$. The \mathcal{L} -subsumption query, instance query, and conjunctive query differences between \mathcal{T}_1 and \mathcal{T}_2 w.r.t. Σ are the sets $\text{cDiff}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2)$, $\text{iDiff}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2)$ and $\text{qDiff}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2)$, where

- $\varphi \in \text{cDiff}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2)$ iff φ is a \mathcal{L} -inclusion with $\text{sig}(\varphi) \subseteq \Sigma$, $\mathcal{T}_1 \models \varphi$ and $\mathcal{T}_2 \not\models \varphi$;
- $(\mathcal{A}, \lambda) \in \text{iDiff}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2)$ iff \mathcal{A} is an ABox with $\text{sig}(\mathcal{A}) \cap (N_C \cup N_R) \subseteq \Sigma$ and λ a Σ -instance assertion, $(\mathcal{T}_1, \mathcal{A}) \models \lambda$ and $(\mathcal{T}_2, \mathcal{A}) \not\models \lambda$;
- $(\mathcal{A}, q(x)) \in \text{qDiff}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2)$ iff \mathcal{A} is an ABox with $\text{sig}(\mathcal{A}) \cap (N_C \cup N_R) \subseteq \Sigma$ and $q(x)$ a Σ -conjunctive query, $(\mathcal{T}_1, \mathcal{A}) \models q(x)$ and $(\mathcal{T}_2, \mathcal{A}) \not\models q(x)$.

The notion of subsumption, instance, and conjunctive query difference between \mathcal{T}_1 and \mathcal{T}_2 w.r.t. Σ is not symmetric. In general, e.g., for subsumption query difference, it holds that $\text{cDiff}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2) \neq \text{cDiff}_{\Sigma}(\mathcal{T}_2, \mathcal{T}_1)$. Moreover, $\text{cDiff}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$ does not imply $\text{cDiff}_{\Sigma}(\mathcal{T}_2, \mathcal{T}_1) = \emptyset$, and vice versa. The same holds for instance and conjunctive query difference.

In remainder of the thesis, if \mathcal{L} is not specified in notions as in $\text{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$, then \mathcal{L} is taken to be the language in which \mathcal{T}_1 is formulated.

The following examples illustrate subsumption, instance, and conjunctive query differences.

Example 20 Let $\mathcal{T}_1 = \{A \equiv \exists r.B \sqcap X, r \sqsubseteq s\}$, $\mathcal{T}_2 = \{A \equiv \exists s.B \sqcap X, r \sqsubseteq s\}$ and $\Sigma = \{A, B, r\}$. Then it holds that $A \sqsubseteq X \notin \text{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ as $\mathcal{T}_1 \models A \sqsubseteq X$ and $\mathcal{T}_2 \models A \sqsubseteq X$. We also have that $\mathcal{T}_1 \models \exists r.B \sqcap X \sqsubseteq A$, but $\mathcal{T}_2 \not\models \exists r.B \sqcap X \sqsubseteq A$. Therefore, $\exists r.B \sqcap X \sqsubseteq A \in \text{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$.

Example 21 Let $\mathcal{T}_1 = \{\text{ran}(r) \sqsubseteq A, \text{ran}(s) \sqsubseteq B, X \equiv A \sqcap B\}$, $\mathcal{T}_2 = \emptyset$ and $\Sigma = \{X, s, r\}$. We have that the inclusion $\varphi = \text{ran}(r) \sqcap \text{ran}(s) \sqsubseteq X$ uses only symbols from Σ and that φ is entailed by \mathcal{T}_1 but not by \mathcal{T}_2 . Note, however, that φ is not an \mathcal{ELH}^r -inclusion according to Definition 4. Moreover, \mathcal{ELH}^r -inclusions such as $\text{ran}(r) \sqsubseteq A$ and $\text{ran}(s) \sqsubseteq B$ are not \mathcal{ELH}_Σ^r -inclusions as $\text{sig}(\text{ran}(r) \sqsubseteq A) \not\subseteq \Sigma$ and $\text{sig}(\text{ran}(s) \sqsubseteq B) \not\subseteq \Sigma$. In fact, it holds that $\text{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$. Extending Σ to, say, the signature $\Sigma' = \Sigma \cup \{A\}$ yields $\text{cDiff}_{\Sigma'}(\mathcal{T}_1, \mathcal{T}_2) \neq \emptyset$ as $\mathcal{T}_1 \models \text{ran}(r) \sqsubseteq A$ and $\mathcal{T}_2 \not\models \text{ran}(r) \sqsubseteq A$. Additionally, let $\mathcal{A} = \{r(a, c), s(b, c)\}$ be a *ABox*. It can readily be verified that $(\mathcal{T}_1, \mathcal{A}) \models X(c)$ and $(\mathcal{T}_2, \mathcal{A}) \not\models X(c)$. So, we have that $(\mathcal{A}, X(c)) \in \text{iDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$.

Example 22 Let $\mathcal{T}_1 = \{A \sqsubseteq \exists r.\top, r \sqsubseteq s, r \sqsubseteq t\}$, $\mathcal{T}_2 = \{A \sqsubseteq \exists s.\top \sqcap \exists t.\top\}$ and $\Sigma = \{A, s, t\}$. We have that, e.g., $A \sqsubseteq \exists s.\top$ and $A \sqsubseteq \exists t.\top$ are each entailed by \mathcal{T}_1 and \mathcal{T}_2 . This holds for all \mathcal{ELH}_Σ^r -inclusions. Note that $r \sqsubseteq s$ and $r \sqsubseteq t$ are not \mathcal{ELH}_Σ^r -inclusions as $\text{sig}(r \sqsubseteq s) \not\subseteq \Sigma$ and $\text{sig}(r \sqsubseteq t) \not\subseteq \Sigma$. In fact, we have that $\text{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$. Additionally, let $\mathcal{A}_1 = \{r(a, b)\}$ be an *ABox* and $\lambda_1 = s(a, b)$ be an assertion. We have that λ_1 follows from $(\mathcal{T}_1, \mathcal{A}_1)$ but not from $(\mathcal{T}_2, \mathcal{A}_1)$. However, \mathcal{A}_1 is an *ABox* that is not of the required form as it contains a role name that is not in Σ , i.e., $\text{sig}(\mathcal{A}_1) \cap (\text{N}_C \cup \text{N}_R) \not\subseteq \Sigma$. In fact, for *ABoxes* \mathcal{A} satisfying $\text{sig}(\mathcal{A}) \cap (\text{N}_C \cup \text{N}_R) \subseteq \Sigma$ and for Σ -instance queries λ , it holds that $(\mathcal{T}_1, \mathcal{A}) \models \lambda$ and $(\mathcal{T}_2, \mathcal{A}) \models \lambda$. That is, $\text{iDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$. Finally, let $\mathcal{A}_2 = \{A(a)\}$ be an *ABox* and $q(x) = \exists x.(s(a, x) \wedge t(a, x))$ be a conjunctive query (with no free variables). We have that $(\mathcal{T}_1, \mathcal{A}_2) \models q()$ but $(\mathcal{T}_2, \mathcal{A}_2) \not\models q(x)$. So, $(\mathcal{A}_2, q(x)) \in \text{qDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$.

If the set $\text{cDiff}_\Sigma^\mathcal{L}(\mathcal{T}_1, \mathcal{T}_2)$ is not empty, then it typically contains infinitely many concept inclusions. We make use of the *primitive witnesses theorems* from [23], which state that if there is a concept inclusion difference in $\text{cDiff}_\Sigma^\mathcal{L}(\mathcal{T}_1, \mathcal{T}_2)$, then there exists an inclusion in $\text{cDiff}_\Sigma^\mathcal{L}(\mathcal{T}_1, \mathcal{T}_2)$ of one of the following five types $\delta_1, \dots, \delta_5$, which are built according to the grammar rules below:

$$\begin{aligned} \delta_1 & ::= r \sqsubseteq s \\ \delta_2 & ::= C \sqsubseteq A \mid \text{ran}(r) \sqcap C \sqsubseteq A \\ \delta_3 & ::= D \sqsubseteq A \\ \delta_4 & ::= A \sqsubseteq C \mid \text{dom}(r) \sqsubseteq C \mid \text{ran}(r) \sqsubseteq C \\ \delta_5 & ::= A \sqsubseteq E \mid \text{dom}(r) \sqsubseteq E \mid \text{ran}(r) \sqsubseteq E \end{aligned}$$

where δ_1 ranges over role inclusions, δ_2 and δ_4 are \mathcal{ELH}^r -inclusions, δ_3 is an $\mathcal{EL}^{\text{ran}}$ -inclusion and δ_5 is an $\mathcal{EL}^{\text{ran}, \sqcap, \sqcup}$ -inclusion. Note that each of these inclusions has either a simple left-hand or right-hand side.

The following table summarises results from [23], which identify the types of inclusions that are sufficient to represent the concept inclusion difference between \mathcal{ELH}^r -terminologies for the three query languages that we consider (and from now on simply refer to as ‘ran’, ‘Ran’ and ‘Ran $^{\square,u}$ ’).²

Query language	Abbreviation	Types of inclusions in $\text{cDiff}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2)$
\mathcal{ELH}_{Σ}^r	ran	$\delta_1, \delta_2, \delta_4$
$\mathcal{EL}_{\Sigma}^{\text{ran}}$	Ran	$\delta_1, \delta_3, \delta_4$
$\mathcal{EL}_{\Sigma}^{\text{ran}, \square, u}$	Ran $^{\square, u}$	$\delta_1, \delta_3, \delta_5$

Figure 2.1 The concept differences between \mathcal{ELH}^r -terminologies

In case two ontologies are logically different, the set $\text{cDiff}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2)$ consists of infinitely many concept inclusions. The *primitive witnesses theorems* from [23] allow us to consider only certain inclusions of a simpler syntactic form.

Definition 23 (Subsumption Difference Witnesses) *Let \mathcal{T}_1 and \mathcal{T}_2 be \mathcal{ELH}^r -terminologies, Σ be a signature and $\mathcal{L} \in \{\text{ran}, \text{Ran}, \text{Ran}^{\square, u}\}$. The set of all \mathcal{L} -subsumption difference witnesses, denoted by $\text{Wtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2)$, is the set of $\phi \in \{A, \text{dom}(r), \text{ran}(r), r \mid A \in \Sigma \cap \text{N}_{\mathbb{C}}, r \in \Sigma \cap \text{N}_{\mathbb{R}}\}$ such that ϕ occurs either on the left-hand side or right-hand side of \mathcal{L} -concept inclusions α where $\mathcal{T}_1 \models \alpha$, but $\mathcal{T}_2 \not\models \alpha$.*

We use $\text{lhs}(\alpha)$ (resp. $\text{rhs}(\alpha)$) to represent the left-hand side (resp. right-hand side) of an inclusion α .

Theorem 24 *Let \mathcal{T}_1 and \mathcal{T}_2 be \mathcal{ELH}^r -terminologies, Σ be a signature and $\mathcal{L} \in \{\text{ran}, \text{Ran}, \text{Ran}^{\square, u}\}$. The set of all \mathcal{L} -subsumption difference witnesses is given by*

$$\text{Wtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2) = \text{roleWtn}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2) \cup \text{lhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2) \cup \text{rhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2),$$

where the set $\text{roleWtn}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2) = \{r \in \Sigma \cap \text{N}_{\mathbb{R}} \mid r \sqsubseteq s \text{ or } s \sqsubseteq r \in \text{the set } \text{cDiff}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2)\}$, the set $\text{lhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2) = \{\phi \in (\Sigma \cap \text{N}_{\mathbb{C}}) \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}} \mid \phi \sqsubseteq \text{rhs}(\alpha) \text{ and } \alpha \text{ is a type-}\delta_4 \text{ or type-}\delta_5 \text{ inclusions in the set } \text{cDiff}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2)\}$ and the set $\text{rhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2) = \{A \in \Sigma \cap \text{N}_{\mathbb{C}} \mid \text{lhs}(\alpha) \sqsubseteq A \text{ and } \alpha \text{ is a type-}\delta_2 \text{ or type-}\delta_3 \text{ inclusions in the set } \text{cDiff}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2)\}$ respectively, depending on the query language \mathcal{L} (cf. Fig. 2.1). The set $\text{Wtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2)$ can be seen as a finite representation of the set $\text{cDiff}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2)$, which is typically infinite when it is not empty.

As a corollary of the primitive witnesses theorems in [23], we have that the representation is complete in the following sense: $\text{cDiff}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$ iff $\text{roleWtn}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2) = \text{lhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2) =$

²We refer to Theorems 40 and 61 in [23].

$\text{rhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$. Thus, for deciding the existence of concept inclusion differences, it is equivalent to decide non-emptiness of one of the three witnesses sets.

Observe that the set $\text{Wtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2)$ is finite as Σ is finite. Consequently, it can be seen as a succinct representation of the set $\text{cDiff}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2)$ in the sense that:

$$\text{cDiff}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset \text{ iff } \text{Wtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset \text{ [23].}$$

Example 25 Let \mathcal{T} consist of the following four axioms:

$$\begin{array}{ll} \alpha_1 : A \sqsubseteq B \sqcap \exists r.X & \alpha_2 : B \sqsubseteq A \\ \alpha_3 : X \equiv A \sqcap B & \alpha_4 : Y \equiv B \sqcap \exists r.(X \sqcap \exists s.A) \end{array}$$

For $\Sigma = \{A, B\}$, it holds that $\text{Wtn}_{\Sigma}(\mathcal{T}, \{\alpha_1, \alpha_2\}) = \text{cDiff}_{\Sigma}(\mathcal{T}, \{\alpha_1, \alpha_2\}) = \emptyset$ and $\text{Wtn}_{\Sigma}(\mathcal{T}, \emptyset) = \Sigma$ as $A \sqsubseteq B, B \sqsubseteq A \in \text{cDiff}_{\Sigma}(\mathcal{T}, \emptyset)$. If $\Sigma = \{A, r\}$, we have that $\text{Wtn}_{\Sigma}(\mathcal{T}, \mathcal{T} \setminus \{\alpha_1\}) = \{A\}$ as $A \sqsubseteq \exists r.\top \in \text{cDiff}_{\Sigma}(\mathcal{T}, \mathcal{T} \setminus \{\alpha_1\})$.

As we can see from Definition 23, the set $\text{Wtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2)$ can be seen as a finite representation of the set $\text{cDiff}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2)$ [23], which is typically infinite.

The following theorem states that $\mathcal{E}\mathcal{L}^{\text{ran}}$ -subsumption and $\mathcal{E}\mathcal{L}^{\text{ran}, \sqcap, \sqcup}$ -subsumption queries are sufficient to capture the absence of instance query and conjunctive query differences.

Theorem 26 Let \mathcal{T}_1 and \mathcal{T}_2 be $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -TBoxes and let Σ be a signature. Then:

- $\text{rhsWtn}_{\Sigma}^{\text{ran}}(\mathcal{T}_1, \mathcal{T}_2) \neq \text{rhsWtn}_{\Sigma}^{\text{Ran}}(\mathcal{T}_1, \mathcal{T}_2)$
- $\text{rhsWtn}_{\Sigma}^{\text{Ran}}(\mathcal{T}_1, \mathcal{T}_2) = \text{rhsWtn}_{\Sigma}^{\text{Ran}, \sqcap, \sqcup}(\mathcal{T}_1, \mathcal{T}_2)$
- $\text{lhsWtn}_{\Sigma}^{\text{ran}}(\mathcal{T}_1, \mathcal{T}_2) = \text{lhsWtn}_{\Sigma}^{\text{Ran}}(\mathcal{T}_1, \mathcal{T}_2)$
- $\text{lhsWtn}_{\Sigma}^{\text{Ran}}(\mathcal{T}_1, \mathcal{T}_2) \neq \text{lhsWtn}_{\Sigma}^{\text{Ran}, \sqcap, \sqcup}(\mathcal{T}_1, \mathcal{T}_2)$

This theorem follows from Lemmas 62 and 63 in [23]. The following lemma shows that logical difference has the property of monotonicity.

Lemma 27 (Monotonicity of Logical Difference) Let $\mathcal{T}_1, \mathcal{T}_2$ be two $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies, and $\mathcal{T}'_2 \subseteq \mathcal{T}_2$. Additionally, let Σ be a signature. The following statements hold:

- $\text{cDiff}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2) \subseteq \text{cDiff}_{\Sigma}(\mathcal{T}_1, \mathcal{T}'_2)$;
- $\text{iDiff}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2) \subseteq \text{iDiff}_{\Sigma}(\mathcal{T}_1, \mathcal{T}'_2)$;
- $\text{qDiff}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2) \subseteq \text{qDiff}_{\Sigma}(\mathcal{T}_1, \mathcal{T}'_2)$;
- $\text{Wtn}_{\Sigma}(\mathcal{T}_1, \mathcal{T}_2) \subseteq \text{Wtn}_{\Sigma}(\mathcal{T}_1, \mathcal{T}'_2)$.

Algorithms for computing the witnesses sets, and hence for deciding whether a logical difference w.r.t. a signature exists, have been implemented in the CEX2.5 tool.³ Given two acyclic \mathcal{ELH}^r -terminologies \mathcal{T}_1 and \mathcal{T}_2 and a signature Σ as input, CEX2.5 can compute and output the set $\text{Wtn}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ in a fully automatic way.

We still note that a new approach for computing logical differences that can also handle large cyclic terminologies has recently been introduced [12, 34].

2.2.1 Proof system for \mathcal{ELH}^r

A concept name A is called *pseudo-primitive* in a normalised terminology \mathcal{T} if it does not occur on the left-hand side of an inclusion of the form $A \equiv C$ in \mathcal{T} . Note that concept names that do not occur in \mathcal{T} are pseudo-primitive in \mathcal{T} . A concept name A is said to be *non-conjunctive* in \mathcal{T} if it is pseudo-primitive in \mathcal{T} or there exists a concept of the form $\exists r.C$ such that $A \equiv \exists r.C \in \mathcal{T}$. Otherwise, A is called *conjunctive* in \mathcal{T} . Thus, A is conjunctive in \mathcal{T} iff there exist concept names B_1, \dots, B_n , $n \geq 2$ such that $A \equiv B_1 \sqcap \dots \sqcap B_n \in \mathcal{T}$.

We can now categorise a concept name A w.r.t. a normalised terminology \mathcal{T} as one of the following holds:

- A is pseudo-primitive in \mathcal{T} ,
- A is conjunctive in \mathcal{T} , or
- there exists exactly one axiom of the form $A \equiv \exists r.B$ in \mathcal{T} .

Definition 28 Let C be an \mathcal{EL} -concept. We define the set $\text{sub}(C) \subseteq \mathcal{EL}$ inductively as follows:

- $\text{sub}(\top) = \{\top\}$;
- $\text{sub}(A) = \{A\}$;
- $\text{sub}(\exists r.C) = \{\exists r.C\} \cup \text{sub}(C)$;
- $\text{sub}(C_1 \sqcap C_2) = \{C_1 \sqcap C_2\} \cup \text{sub}(C_1) \cup \text{sub}(C_2)$.

Moreover, we define the set $\text{sub}_0(C) \subseteq \mathcal{ELH}$ inductively as follows

- $\text{sub}_0(\top) = \{\top\}$;
- $\text{sub}_0(A) = \{A\}$;
- $\text{sub}_0(\exists r.C) = \{\exists r.C\}$;
- $\text{sub}_0(C_1 \sqcap C_2) = \text{sub}_0(C_1) \cup \text{sub}_0(C_2)$.

$$\begin{array}{c}
\overline{C \sqsubseteq C} \text{ (AX)} \quad \overline{C \sqsubseteq \top} \text{ (AXTOP)} \quad \frac{C \sqsubseteq D}{\exists r.C \sqsubseteq \exists r.D} \text{ (EX)} \quad \frac{\exists r.(C \sqcap \text{ran}(r)) \sqsubseteq D}{\exists r.C \sqsubseteq D} \text{ (EXRAN)} \\
\\
\frac{C \sqsubseteq E}{C \sqcap D_1 \sqcap \dots \sqcap D_n \sqsubseteq E} \text{ (ANDL)} \quad \frac{C \sqsubseteq E_1 \dots C \sqsubseteq E_n}{C \sqsubseteq E_1 \sqcap \dots \sqcap E_n} \text{ (ANDR)} \\
\\
\frac{C_A \sqsubseteq D}{A \sqsubseteq D} \text{ (DEF L)} \quad \frac{D \sqsubseteq C_A}{D \sqsubseteq A} \text{ (DEF R)} \quad \text{where } A \equiv C_A \in \mathcal{T} \\
\\
\frac{C_A \sqsubseteq D}{A \sqsubseteq D} \text{ (PDEF L)} \quad \text{where } A \sqsubseteq C_A \in \mathcal{T} \\
\\
\frac{B \sqsubseteq D}{\exists r.C \sqsubseteq D} \text{ (DOM)} \quad \text{where } \text{dom}(r) \sqsubseteq B \in \mathcal{T} \\
\\
\frac{A \sqsubseteq D}{\text{ran}(r) \sqsubseteq D} \text{ (RAN)} \quad \text{where } \text{ran}(r) \sqsubseteq A \in \mathcal{T} \\
\\
\frac{\exists s.C \sqsubseteq D}{\exists r.C \sqsubseteq D} \text{ (SUB)} \quad \frac{\text{ran}(s) \sqsubseteq D}{\text{ran}(r) \sqsubseteq D} \text{ (RANSUB)} \quad \text{where } r \sqsubseteq s \in \mathcal{T}
\end{array}$$

Figure 2.2 Gentzen-style proof system for normalised \mathcal{ELH}^r -terminologies

Figure 2.2 depicts the Gentzen-style proof system for \mathcal{ELH}^r -terminologies from [17] extended with a number of inference rules to deal with the top-concept, and concept inclusions of the form $A \sqsubseteq C$, domain- and range restrictions, and role inclusions [23]. For a normalised \mathcal{ELH}^r -terminology \mathcal{T} and concepts C, D , we write $C \vdash_{\mathcal{T}} D$ iff there exists a proof tree of $C \sqsubseteq D$ using the inference rules of the calculus shown in Figure 2.2. We use this to characterise which concept inclusions are logically entailed by a normalised terminology \mathcal{T} .

Example 29 Let $\mathcal{T} = \{A \equiv B_1 \sqcap X, X \equiv \exists r.B_2\}$. Then a proof tree for $\mathcal{T} \models B_1 \sqcap \exists r.B_2 \sqsubseteq A$ looks as follows.

$$\frac{\frac{\frac{\overline{B_1 \sqsubseteq B_1} \text{ (AX)}}{B_1 \sqcap \exists r.B_2 \sqsubseteq B_1} \text{ (ANDL)} \quad \frac{\frac{\overline{\exists r.B_2 \sqsubseteq \exists r.B_2} \text{ (AX)}}{\exists r.B_2 \sqsubseteq X} \text{ (DEF R)}}{B_1 \sqcap \exists r.B_2 \sqsubseteq X} \text{ (ANDL)}}{B_1 \sqcap \exists r.B_2 \sqsubseteq B_1 \sqcap X} \text{ (ANDR)}}{B_1 \sqcap \exists r.B_2 \sqsubseteq A} \text{ (DEF R)}$$

The deduction system is sound and complete w.r.t. the model theoretic semantics.

Theorem 30 (see [17, 23]) *Let \mathcal{T} be a normalised \mathcal{ELH}^r -terminology and let C, D be $\mathcal{EL}^{\text{ran}}$ -concepts. Then: $\mathcal{T} \models C \sqsubseteq D$ iff $C \vdash_{\mathcal{T}} D$.*

It follows from Theorem 30 that $\mathcal{T} \not\models \top \sqsubseteq A$ holds for every concept name A and for every normalised \mathcal{ELH}^r -terminology \mathcal{T} .

³The tool is available under an open-source license from <https://cgi.csc.liv.ac.uk/~konev/software/>

2.2.2 Properties Regarding Logical Differences

The following lemma allows us to characterise an entailment of the form $\mathcal{T} \models C \sqsubseteq A$ in terms of the conjuncts that occur in C .

Lemma 31 *Let \mathcal{T} be a normalised \mathcal{ELH} -terminology, r a role name, A a concept name and D an \mathcal{EL} -concept.*

1. Assume $\mathcal{T} \models C \sqsubseteq A$, where

$$C = \prod_{1 \leq i \leq n} A_i \sqcap \prod_{1 \leq j \leq m} \exists r_j.C_j,$$

A is pseudo-primitive in \mathcal{T} , A_i are concept names for $1 \leq i \leq n$, C_j are \mathcal{EL} -concepts for $1 \leq j \leq m$, and $m, n \geq 0$. Then there exists A_i , $1 \leq i \leq n$, such that $\mathcal{T} \models A_i \sqsubseteq A$.

2. Assume now $\mathcal{T} \models C \sqsubseteq \exists r.D$, where

$$C = \prod_{1 \leq i \leq n} A_i \sqcap \prod_{1 \leq j \leq m} \exists r_j.C_j,$$

A_i are concept names for $1 \leq i \leq n$, C_j are \mathcal{EL} -concepts for $1 \leq j \leq m$, and $m, n \geq 0$. Then either

- there exists A_i , $1 \leq i \leq n$, such that $\mathcal{T} \models A_i \sqsubseteq \exists r.D$, or
- there exists r_j , $1 \leq j \leq m$, such that $r_j = r$ and $\mathcal{T} \models C_j \sqsubseteq D$.

Lemma 31 is an extension of Lemma 15 in [23] additionally allowing for role inclusions in the terminology. The proof of Lemma 31 uses Theorem 30, which states soundness and completeness of the Gentzen-style proof system for \mathcal{ELH}^r -terminologies in Figure 2.2. The proof proceeds by induction on the length of the left-hand side of a subsumption (cf. Item 1 and 2 of Lemma 31) that follows from the terminology. The idea is to analyse possible derivations for the subsumption (determining which inference rules could have been used last) and show that a derivation for a simpler subsumption can be found. Role inclusion axioms are accounted for by the inference rule (SUB) in Figure 2.2. Since neither \mathcal{ELH} -terminologies nor the subsumptions considered in Lemma 31 contain range concepts, the inference rules (EXRAN), (RAN) and (RANSUB) do not apply.

Lemma 32 *Let \mathcal{T} be a normalised \mathcal{ELH} -terminology, let X be a concept name, let $\exists r.D$ be an \mathcal{ELH} -concept such that $\mathcal{T} \models X \sqsubseteq \exists r.D$. Then there exists an axiom $Y \bowtie \exists s.Z \in \mathcal{T}$ ($\bowtie \in \{\sqsubseteq, \equiv\}$) such that $\mathcal{T} \models X \sqsubseteq Y$, $\mathcal{T} \models s \sqsubseteq r$ and $\mathcal{T} \models Z \sqsubseteq D$.*

Proof Let Δ be the derivation of the inclusion $X \sqsubseteq \exists r.D$ w.r.t. \mathcal{T} . We assume now towards a contradiction that there does not exist an application of the (DEFL), (PDEFL) or (SUB) rule w.r.t. an axiom $Y \bowtie \exists r.Z \in \mathcal{T}$ in Δ . It is then easy to see that no concept of the form $\exists r.E$ occurs on the

left-hand side of any inclusion in Δ . We can infer that no concept of the form $\exists r.E$ occurs on the right-hand side of any inclusion in Δ as well, i.e. Δ is not a derivation of the inclusion $A \sqsubseteq \exists r.D$, which contradicts our assumption. \square

Definition 33 (Concept Witness [23]) Let \mathcal{T}_1 and \mathcal{T}_2 be normalised \mathcal{ELH} -terminologies, and let Σ be a signature. The set $\text{cWtn}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$ consists of concept names A from Σ such that there exists an \mathcal{EL}_Σ -concept inclusion α of the form $A \sqsubseteq D$ or $C \sqsubseteq A$ satisfying $\mathcal{T}_1 \models \alpha$ but $\mathcal{T}_2 \not\models \alpha$.

Lemma 34 (see [23]) Let \mathcal{T} be an \mathcal{ELH}^r -terminology, A_j concept names and $\exists r.D$ an \mathcal{EL} -concept. Assume

$$\mathcal{T} \models \prod_{1 \leq i \leq l} \text{ran}(s_i) \sqcap \prod_{1 \leq j \leq n} A_j \sqcap \prod_{1 \leq k \leq m} \exists r_k.C_k \sqsubseteq \exists r.D,$$

where C_k , $1 \leq k \leq m$, are $\mathcal{EL}^{\text{ran}}$ -concepts and $l, m, n \geq 0$. Then at least one of the following conditions holds:

- (e1) there exists r_k , $1 \leq k \leq m$, such that $\mathcal{T} \models r_k \sqsubseteq r$ and $\mathcal{T} \models C_k \sqcap \text{ran}(r_k) \sqsubseteq D$;
- (e2) there exists A_j , $1 \leq j \leq n$, such that $\mathcal{T} \models A_j \sqsubseteq \exists r.D$;
- (e3) there exists r_k , $1 \leq k \leq m$, such that $\mathcal{T} \models \text{dom}(r_k) \sqsubseteq \exists r.D$;
- (e4) there exists s_i , $1 \leq i \leq l$, such that $\mathcal{T} \models \text{ran}(s_i) \sqsubseteq \exists r.D$.

Now assume that A is pseudo-primitive and

$$\mathcal{T} \models \prod_{1 \leq i \leq l} \text{ran}(s_i) \sqcap \prod_{1 \leq j \leq n} A_j \sqcap \prod_{1 \leq k \leq m} \exists r_k.C_k \sqsubseteq A,$$

where C_k , $1 \leq k \leq m$, are $\mathcal{EL}^{\text{ran}}$ -concepts and $l, m, n \geq 0$. Then at least one of the following conditions holds:

- (a1) there exists A_j , $1 \leq j \leq n$ such that $\mathcal{T} \models A_j \sqsubseteq A$;
- (a2) there exists r_k , $1 \leq k \leq m$ such that $\mathcal{T} \models \text{dom}(r_k) \sqsubseteq A$;
- (a3) there exists s_i , $1 \leq i \leq l$ such that $\mathcal{T} \models \text{ran}(s_i) \sqsubseteq A$.

The set of $\mathcal{EL}^{\text{ran}, \sqcap}$ -concepts is defined as the set of all $\mathcal{EL}^{\text{ran}, \sqcap, u}$ -concepts without the universal role.

Lemma 35 (see [23]) Let \mathcal{T} be an \mathcal{ELH}^r -terminology and $\exists R.D$ a $\mathcal{EL}^{\text{ran}, \sqcap}$ -concept with $R = t_1 \sqcap \dots \sqcap t_q$ a conjunction of role names. Assume

$$\mathcal{T} \models \prod_{1 \leq i \leq l} \text{ran}(s_i) \sqcap \prod_{1 \leq j \leq n} A_j \sqcap \prod_{1 \leq k \leq m} \exists r_k.C_k \sqsubseteq \exists R.D,$$

where C_k , $1 \leq k \leq m$, are $\mathcal{E}\mathcal{L}^{\text{ran}}$ -concepts and $l, m, n \geq 0$. Then at least one of the following conditions holds:

- (e1 $_{\sqcap}$) there exists r_k , $1 \leq k \leq m$, such that $r_k \sqsubseteq_{\mathcal{T}} t_1, \dots, r_k \sqsubseteq_{\mathcal{T}} t_q$, and $\mathcal{T} \models C_k \sqcap \text{ran}(r_k) \sqsubseteq D$;
- (e2 $_{\sqcap}$) there exists A_j , $1 \leq j \leq n$, such that $\mathcal{T} \models A_j \sqsubseteq \exists R.D$;
- (e3 $_{\sqcap}$) there exists r_k , $1 \leq k \leq m$, such that $\mathcal{T} \models \text{dom}(r_k) \sqsubseteq \exists R.D$;
- (e4 $_{\sqcap}$) there exists s_i , $1 \leq i \leq l$, such that $\mathcal{T} \models \text{ran}(s_i) \sqsubseteq \exists R.D$.

If u is the universal role and $\mathcal{T} \models C \sqsubseteq \exists u.D$, where C is a $\mathcal{E}\mathcal{L}^{\text{ran}}$ -concept and D is a $\mathcal{E}\mathcal{L}^{\text{ran}, \sqcap}$ -concept, then at least one of the following holds:

- (e1 $_u$) there exists a subconcept $\exists r.C'$ of C such that $\mathcal{T} \models C' \sqcap \text{ran}(r) \sqsubseteq D$;
- (e2 $_u$) there exists a concept name A in C such that $\mathcal{T} \models A \sqsubseteq \exists u.D$;
- (e3 $_u$) there exists a role name r in C such that $\mathcal{T} \models \text{dom}(r) \sqsubseteq \exists u.D$;
- (e4 $_u$) there exists a role name r in C such that $\mathcal{T} \models \text{ran}(r) \sqsubseteq \exists u.D$;
- (e5 $_u$) $\mathcal{T} \models C \sqsubseteq D$;
- (e6 $_u$) there exists a subconcept $(\text{ran}(r) \sqcap C')$ of C such that $\mathcal{T} \models \exists r.C' \sqsubseteq D$.

2.3 Ontology Modules

A module is a subset of an ontology that captures the knowledge that the ontology describes about a signature. In general, there are two different kinds of ontology modules. One is semantic modules that are based on model-theoretic inseparability [27] and the other one is locality based modules that are based on syntactic check [42].

2.3.1 Inseparability and Semantic Modules

In this section, we review some notions about inseparability and relative properties, which are introduced in [27, 40].

Definition 36 (Σ -Reduct) Let \mathcal{I} be an interpretation and Σ a signature. The Σ -reduct $\mathcal{I}|_{\Sigma}$ of \mathcal{I} is the interpretation obtained from \mathcal{I} by the following rules:

- $\Delta^{\mathcal{I}|_{\Sigma}} := \Delta^{\mathcal{I}}$;
- $\sigma^{\mathcal{I}|_{\Sigma}} := \sigma^{\mathcal{I}}$, for all $\sigma \in \Sigma$;
- $\sigma^{\mathcal{I}|_{\Sigma}} := \emptyset$, for all $\sigma \in (N_C \cup N_R) \setminus \Sigma$.

Definition 37 (Σ -Inseparability) Let \mathcal{T}_1 and \mathcal{T}_2 be general TBoxes and Σ a signature. Then \mathcal{T}_1 and \mathcal{T}_2 are Σ -inseparable, denoted as $\mathcal{T}_1 \equiv_{\Sigma} \mathcal{T}_2$, iff $\{\mathcal{I} |_{\Sigma} \mid \mathcal{I} \models \mathcal{T}_1\} = \{\mathcal{I} |_{\Sigma} \mid \mathcal{I} \models \mathcal{T}_2\}$.

Example 38 Considering the following terminologies \mathcal{T}_1 , \mathcal{T}_2 and \mathcal{T}_3 . \mathcal{T}_1 consists of the following axiom (FS is short for finding_site(attribute)):

$$\begin{aligned} \text{Disorder_of_Body_Cavity}(\text{disorder}) &\equiv \text{Disorder_of_Body_Site}(\text{disorder}) \sqcap \\ &\quad \exists \text{FS.Trunk_Structure}(\text{Body_Structure}), \end{aligned}$$

\mathcal{T}_2 consists of the following axioms:

$$\begin{aligned} \text{Disorder_of_Body_Cavity}(\text{disorder}) &\sqsubseteq \text{Disorder_of_trunk}(\text{disorder}) \\ \text{Disorder_of_trunk}(\text{disorder}) &\sqsubseteq \text{Disorder_of_Body_Site}(\text{disorder}) \sqcap \\ &\quad \exists \text{FS.Trunk_Structure}(\text{Body_Structure}), \end{aligned}$$

and \mathcal{T}_3 consists of the following axiom:

$$\text{Disorder_of_Body_Cavity}(\text{disorder}) \sqsubseteq \text{Disorder_of_Body_Site}(\text{disorder}).$$

We have that $\mathcal{T}_1 \not\equiv \mathcal{T}_2 \not\equiv \mathcal{T}_3$. But for

$$\Sigma = \{\text{Disorder_of_Body_Cavity}(\text{disorder}), \text{Disorder_of_Body_Site}(\text{disorder})\}$$

we have that $\mathcal{T}_1 \equiv_{\Sigma} \mathcal{T}_2 \equiv_{\Sigma} \mathcal{T}_3$.

The following lemma shows that inseparability has the monotonicity property.

Lemma 39 Let \mathcal{T}_1 and \mathcal{T}_2 be two general TBoxes. Additionally, let Σ_1 and Σ_2 be two signatures and $\Sigma_1 \subseteq \Sigma_2$. Then $\mathcal{T}_1 \equiv_{\Sigma_2} \mathcal{T}_2$ implies $\mathcal{T}_1 \equiv_{\Sigma_1} \mathcal{T}_2$.

We then introduce some notions of semantic modules.

Definition 40 (Σ -Modules) Let \mathcal{T} be a general TBox and Σ a signature. Then $\mathcal{M} \subseteq \mathcal{T}$ is a:

- plain Σ -module of \mathcal{T} w.r.t. Σ iff $\mathcal{M} \equiv_{\Sigma} \mathcal{T}$;
- self-contained Σ -module of \mathcal{T} w.r.t. Σ iff $\mathcal{M} \equiv_{\Sigma \cup \text{sig}(\mathcal{M})} \mathcal{T}$;
- depleting Σ -module of \mathcal{T} w.r.t. Σ iff $\mathcal{T} \setminus \mathcal{M} \equiv_{\Sigma \cup \text{sig}(\mathcal{M})} \mathbf{0}$.

We say a plain (self-contained, depleting) Σ -module of \mathcal{T} w.r.t. Σ is minimal iff there does not exist an $\mathcal{M}' \subsetneq \mathcal{M}$ such that \mathcal{M}' is a plain (resp. self-contained, depleting) Σ -module of \mathcal{T} w.r.t. Σ .

The following proposition states the relations among these module notions [27].

Proposition 41 *Let \mathcal{T} be a general TBox and Σ a signature. If \mathcal{M} is a self-contained Σ -module of \mathcal{T} , then it is a plain Σ -module. Additionally, if \mathcal{M} is a depleting Σ -module of \mathcal{T} , then it is a self-contained Σ -module.*

An algorithm for computing a unique minimal depleting module of an acyclic \mathcal{ELI} -terminology, an \mathcal{EL} -terminology additionally allowing for inverse roles, has been implemented in the tool MEX.⁴ Recently, new algorithms to compute depleting module for DL-Lite TBox [31] and \mathcal{ALCQI} -TBox [15] have been developed.

2.3.2 Locality-Based Modules

Extracting semantic modules (as presented in the last subsection) from real-world ontologies is computationally difficult. This is why more practical approaches to module extraction have been investigated resulting, most notably, in the module notions that are based on syntactic locality. Syntactic locality modules are approximating semantic module notions from above, i.e., they contain semantic modules but they cannot guarantee minimality w.r.t. set inclusion. Actually syntactic locality-based modules are usually larger than minimal semantic modules. On the other hand, the algorithm for extracting syntactic locality-based modules is rather efficient.

We consider the syntactic locality-based notions of \perp -locality and \top -locality. Intuitively, an axiom α is \perp -local or \top -local w.r.t. a signature Σ if α becomes a tautology after all symbols that are not in Σ have either been replaced with \perp or \top . We now give the formal definition of \perp -locality and \top -locality for an \mathcal{ELH}^r -ontology.

Definition 42 (\perp -locality) *Let C be an \mathcal{EL} concept, $r \in \mathbb{N}_R$, $A^{\notin \Sigma} \in \mathbb{N}_C \setminus \Sigma$ and $s^{\notin \Sigma} \in \mathbb{N}_R \setminus \Sigma$. Additionally, let C^\perp and C^\top be defined recursively by the following rules:*

$$\begin{aligned} C^\perp &::= A^{\notin \Sigma} \mid \perp \mid C \sqcap C^\perp \mid C^\perp \sqcap C \mid \exists s^{\notin \Sigma}. C \mid \exists r. C^\perp \mid \text{dom}(s^{\notin \Sigma}) \mid \text{ran}(s^{\notin \Sigma}) \\ C^\top &::= \top \mid C^\top \sqcap C^\top \end{aligned}$$

An axiom α is \perp -local w.r.t. a signature Σ if α is of one of the following forms:

$$C^\perp \sqsubseteq C \quad \text{or} \quad C \sqsubseteq C^\top \quad \text{or} \quad C^\top \equiv C^\top \quad \text{or} \quad C^\perp \equiv C^\perp.$$

Definition 43 (\top -locality) *Let C be an \mathcal{EL} concept, $r \in \mathbb{N}_R$, $A^{\notin \Sigma} \in \mathbb{N}_C \setminus \Sigma$ and $s^{\notin \Sigma} \in \mathbb{N}_R \setminus \Sigma$. Additionally, let C^\perp and C^\top be defined recursively by the following rules:*

$$\begin{aligned} C^\perp &::= \perp \mid C \sqcap C^\perp \mid C^\perp \sqcap C \mid \exists r. C^\perp \\ C^\top &::= A^{\notin \Sigma} \mid C^\top \sqcap C^\top \mid \exists s^{\notin \Sigma}. C \mid \text{dom}(s^{\notin \Sigma}) \mid \text{ran}(s^{\notin \Sigma}) \end{aligned}$$

⁴The tool is available under an open-source license from <https://cgi.csc.liv.ac.uk/~konev/software/>

An axiom α is \top -local w.r.t. a signature Σ if α is of one of the following forms:

$$C^\perp \sqsubseteq C \quad \text{or} \quad C \sqsubseteq C^\top \quad \text{or} \quad C^\top \equiv C^\top \quad \text{or} \quad C^\perp \equiv C^\perp.$$

Definition 44 An $\mathcal{EL}\mathcal{H}^r$ -terminology \mathcal{T} is \perp -local (resp. \top -local) w.r.t. Σ iff for every axiom $\alpha \in \mathcal{T}$, α is \perp -local (resp. \top -local) w.r.t. Σ . Generally, we say that \mathcal{T} is local w.r.t. Σ iff \mathcal{T} is either \top -local or \perp -local w.r.t. Σ .

Definition 45 (Locality-based Module) Let \mathcal{T} be an $\mathcal{EL}\mathcal{H}^r$ -terminology and Σ be a signature. A \perp -local (resp. \top -local) module of \mathcal{T} w.r.t. Σ , denoted by \mathcal{M}^\perp (resp. \mathcal{M}^\top), is the minimal $\mathcal{M} \subseteq \mathcal{T}$ such that $\mathcal{T} \setminus \mathcal{M}$ is \perp -local (resp. \top -local) w.r.t. $\Sigma \cup \text{sig}(\mathcal{M})$.

Algorithm 1 [16] shows how to compute \perp -local (resp. \top -local) module of a \mathcal{T} w.r.t. Σ .

Algorithm 1: Computing a Locality-Based Module for a Signature

```

1 function LOCALITY-BASED-MODULE ( $\mathcal{T}, \Sigma, x$ )
2    $\mathcal{M} := \emptyset, \mathcal{T}' := \mathcal{T}$ 
3   repeat
4      $\mathcal{M}' := \mathcal{M}$ 
5     for every  $\alpha \in \mathcal{T}'$  do
6       if  $\alpha$  is not  $x$ -local w.r.t.  $\Sigma \cup \text{sig}(\mathcal{M})$  then
7          $\mathcal{M} := \mathcal{M} \cup \{\alpha\}$ 
8          $\mathcal{T}' := \mathcal{T}' \setminus \{\alpha\}$ 
9       end
10    end
11  until  $\mathcal{M} = \mathcal{M}'$ ;
12  return  $\mathcal{M}$ 

```

Proposition 46 If \mathcal{M} is a \top -local or \perp -local module, then \mathcal{M} is also a plain, self-contained and depleting Σ -module.

As extracting locality-based modules is rather efficient, when computing semantic modules we extract locality-based module first.

2.4 Ontology Justification

In this section, we review the notion of ontology justification.

Definition 47 (Ontology Justification [5]) Let \mathcal{T} be an \mathcal{ELH}^r -TBox and let α be an \mathcal{ELH} -concept inclusion. A subset $\mathcal{M} \subseteq \mathcal{T}$ is called a justification for α from \mathcal{T} iff $\mathcal{M} \models \alpha$ and $\mathcal{M}' \not\models \alpha$ for every $\mathcal{M}' \subsetneq \mathcal{M}$.

We denote the set of all justifications for an \mathcal{ELH} -concept inclusion α from an \mathcal{ELH} -terminology \mathcal{T} with $\text{Just}_{\mathcal{T}}(\alpha)$.

Example 48 Let $\mathcal{T} = \{A \sqsubseteq X_1 \sqcap X_2, X_1 \sqsubseteq Y, X_2 \sqsubseteq Y\}$. Then the set $\text{Just}_{\mathcal{T}}(A \sqsubseteq Y)$ consists of the two sets $\{A \sqsubseteq X_1 \sqcap X_2, X_1 \sqsubseteq Y\}$ and $\{A \sqsubseteq X_1 \sqcap X_2, X_2 \sqsubseteq Y\}$.

Note that $\text{Just}_{\mathcal{T}}(\alpha)$ may contain exponentially many justifications in the number of axioms in \mathcal{T} . Therefore, it is not possible to compute all justifications in polynomial time w.r.t. the size of TBox in every case. But theoretically, it is possible to compute just one justification in polynomial time [5].

2.5 Partial Max-SAT Problem

Maximum Satisfiability (Max-SAT) problem has shown successfully to have many applications in various domains in computer science. In this section, we review relevant knowledge about the Max-SAT problem and its extension, the partial Max-SAT problem.

Let Π be a countably infinite set of propositional variables which we denote with p, q , etc. The set Φ of propositional logic formulas φ is built according to the following grammar rule:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \vee \psi \mid \varphi \rightarrow \psi \mid \varphi \leftrightarrow \psi$$

where $p \in \Pi$ is a propositional variable.

A propositional valuation $v: \Pi \rightarrow \{0, 1\}$ is a function mapping each propositional variable $p \in \Pi$ to a truth value $v(p) \in \{0, 1\}$. The valuation v is extended to propositional formulas as follows:

$$\begin{aligned} v(\neg\varphi) &:= 1 - v(\varphi) \\ v(\varphi \wedge \psi) &:= \min\{v(\varphi), v(\psi)\} \\ v(\varphi \vee \psi) &:= \max\{v(\varphi), v(\psi)\} \\ v(\varphi \rightarrow \psi) &:= \max\{1 - v(\varphi), v(\psi)\} \\ v(\varphi \leftrightarrow \psi) &:= \max\{\min\{v(\varphi), v(\psi)\}, \min\{1 - v(\varphi), 1 - v(\psi)\}\} \end{aligned}$$

where $\varphi, \psi \in \Phi$. A propositional formula $\varphi \in \Phi$ is satisfied in v iff $v(\varphi) = 1$. We say that φ is satisfiable iff there exists a valuation v such that $v(\varphi) = 1$.

A literal ℓ is a propositional variable p or a negated propositional variable $\neg p$, and a clause is a disjunction $\ell_1 \vee \dots \vee \ell_n$ of literals ℓ_i ($0 \leq i \leq n$). For a propositional formula $\varphi \in \Phi$, we denote with $\text{Clauses}(\varphi)$ the transformation of φ into an equi-satisfiable clause set, i.e., $\text{Clauses}(\varphi)$ is a finite set of propositional clauses such that φ is satisfiable iff the formula $\bigwedge_{\psi \in \Psi} \psi$ is satisfiable,

where $\Psi = \text{Clauses}(\varphi)$. It is known that every propositional formula $\varphi \in \Phi$ can be transformed in polynomial time (in the size of φ) into an equi-satisfiable set of clauses. Note that the clauses in $\text{Clauses}(\varphi)$ may contain propositional variables that are not present in φ .

The Boolean Satisfiability Problem (SAT), also known as Propositional Satisfiability Problem, is the problem of deciding whether there exists a valuation that satisfies a given propositional formula φ , or alternatively, all the clauses contained in a set of clauses $\Psi = \text{Clauses}(\varphi)$. As an extension, the set Ψ is split into two sets H and S , called *hard* and *soft* clause sets. The problem of finding a valuation that satisfies all the clauses in H and a maximal number of clauses in S is called the partial Max-SAT problem.

Definition 49 (Partial Max-SAT Problem [14]) *Let H and S be finite sets of propositional clauses. We say that $\mathcal{P} = (H, S)$ is an instance of a partial Max-SAT problem. A valuation v is a solution of \mathcal{P} iff the following conditions are satisfied:*

- (i) $\sum_{\varphi \in H} v(\varphi) = |H|$, and
- (ii) for every propositional valuation v' with $\sum_{\varphi \in H} v'(\varphi) = |H|$ it holds that

$$\sum_{\psi \in S} v'(\psi) \leq \sum_{\psi \in S} v(\psi).$$

In other words, the objective of a partial Max-SAT problem is to find a propositional valuation that satisfies all the hard clauses in H and that satisfies a maximal number of the clauses in S . Note that a partial Max-SAT problem may have several solutions.

Example 50 *Let $\mathcal{P} = (H, S)$ be a partial Max-SAT problem, where $H = \{\neg p_1, p_2 \vee p_3\}$ and $S = \{p_1 \vee \neg p_2, \neg p_3\}$. We have the following two solutions of \mathcal{P} : $v_1(p_1) = 0, v_1(p_2) = 0, v_1(p_3) = 1$ and $v_2(p_1) = 0, v_2(p_2) = 1, v_2(p_3) = 0$.*

A weighted clause is a pair (ψ, w_ψ) , where ψ is a clause and w_ψ is a natural number or infinity meaning the cost for falsifying the clause ψ . If a clause $\psi \in H$, then w_ψ is infinite. Given a valuation v and a set of weighted clauses Φ , the cost of valuation v on Φ is the sum of the weights of the clauses falsified by v . The Weighted Partial Max-SAT problem for weighted clauses is the problem of finding an optimal valuation for the variables of Φ , i.e., a valuation of minimal cost.

Definition 51 (Weighted Partial Max-SAT Problem [14]) *Let H and S be finite sets of weighted propositional clauses of the form (ψ, w_ψ) , where ψ is a clause and w_ψ is a natural number or infinity meaning the cost for falsifying the clause ψ . We say that $\mathcal{P} = (H, S)$ is an instance of a weighted partial Max-SAT problem. A valuation v is a solution of \mathcal{P} iff the following conditions are satisfied:*

- (i) $\sum_{\varphi \in H} v(\varphi) = |H|$, and

(ii) for every propositional valuation v' with $\sum_{\varphi \in H} v'(\varphi) = |H|$ it holds that

$$\sum_{\psi \in S} v'(\psi) \times w_{\psi} \leq \sum_{\psi \in S} v(\psi) \times w_{\psi}.$$

Example 52 Let $\mathcal{P}' = (H, S)$ be a weighted partial Max-SAT problem, where $H = \{(\neg p_1, \infty), (p_2 \vee p_3, \infty)\}$ and $S = \{(p_1 \vee \neg p_2, 4), (\neg p_3, 1)\}$. We have the following solution of \mathcal{P}' : $v_1(p_1) = 0$, $v_1(p_2) = 0$, $v_1(p_3) = 1$. The possible solution of \mathcal{P} in Example 50: $v_2(p_1) = 0$, $v_2(p_2) = 1$, $v_2(p_3) = 0$ is not a solution of \mathcal{P}' , as the cost of valuation v_2 is 4, which is higher than the cost of v_1 .

Chapter 3

Extraction of Ontology Subsumption Justifications

In this chapter, we propose a notion called *subsumption justification* to capture the subsumption knowledge about a term with respect to all primitive and complex concepts built from terms in a given vocabulary Σ .

We proceed by first defining the notions of *subsumer/subsumee module* and *subsumer/subsumee justification* and then present algorithms to compute subsumption justifications in \mathcal{ELH} - and \mathcal{ELH}^r -terminologies. We prove the correctness of algorithms and finally evaluate the algorithms on large biomedical terminologies.

In the rest, we use the operator \otimes to combine sets of axioms as follows. Given a set S and $\mathbb{S}_1, \mathbb{S}_2 \subseteq 2^S$, $\mathbb{S}_1 \otimes \mathbb{S}_2 := \{S_1 \cup S_2 \mid S_1 \in \mathbb{S}_1, S_2 \in \mathbb{S}_2\}$. For instance, if $\mathbb{S}_1 = \{\{\alpha_1, \alpha_2\}, \{\alpha_3\}\}$ and $\mathbb{S}_2 = \{\{\alpha_1, \alpha_3\}, \{\alpha_4, \alpha_5\}\}$, then $\mathbb{S}_1 \otimes \mathbb{S}_2 = \{\{\alpha_1, \alpha_2, \alpha_3\}, \{\alpha_1, \alpha_2, \alpha_4, \alpha_5\}, \{\alpha_3, \alpha_4, \alpha_5\}, \{\alpha_1, \alpha_3\}\}$. For a set \mathbb{M} of sets, we define a function $\text{Minimise}_{\subseteq}(\mathbb{M})$ as follows: $\mathcal{M} \in \text{Minimise}_{\subseteq}(\mathbb{M})$ iff $\mathcal{M} \in \mathbb{M}$ and there does not exist a set $\mathcal{M}' \in \mathbb{M}$ such that $\mathcal{M}' \subsetneq \mathcal{M}$. Continuing the previous example, $\text{Minimise}_{\subseteq}(\mathbb{S}_1 \otimes \mathbb{S}_2) = \{\{\alpha_1, \alpha_3\}, \{\alpha_1, \alpha_2, \alpha_4, \alpha_5\}, \{\alpha_3, \alpha_4, \alpha_5\}\}$. Note that $\mathbb{M} \otimes \{\emptyset\} = \mathbb{M}$, but $\mathbb{M} \otimes \emptyset = \emptyset$. We use \otimes and $\text{Minimise}_{\subseteq}(\cdot)$ to combine sets of subsumer/subsumee/role subsumption modules, whose correctness is guaranteed by Proposition 59.

3.1 Subsumption Justifications of \mathcal{ELH} -Terminologies

In this section we first introduce the definition of (role) subsumption justification based on the notion of (role) subsumption module for an \mathcal{ELH} -terminology and a given signature.

Definition 53 (Subsumption Justification for \mathcal{ELH} -terminology) *Let \mathcal{T} be a normalised \mathcal{ELH} -terminology, let Σ be a signature, and let $X \in \mathbb{N}_{\mathbb{C}}$ and $r \in \mathbb{N}_{\mathbb{R}}$. We say that*

- $\mathcal{M} \subseteq \mathcal{T}$ is an $\langle X, \Sigma \rangle$ -subsumee module of \mathcal{T} iff for every $C \in \mathcal{EL}_\Sigma$, $\mathcal{T} \models C \sqsubseteq X$ implies $\mathcal{M} \models C \sqsubseteq X$;
- $\mathcal{M} \subseteq \mathcal{T}$ is an $\langle X, \Sigma \rangle$ -subsumer module \mathcal{M} of \mathcal{T} iff for every $D \in \mathcal{EL}_\Sigma$, $\mathcal{T} \models X \sqsubseteq D$ implies $\mathcal{M} \models X \sqsubseteq D$;
- $\mathcal{M} \subseteq \mathcal{T}$ is an $\langle r, \Sigma \rangle$ -role subsumption module of \mathcal{T} iff for every $s \in \Sigma$, $\mathcal{T} \models r \sqsubseteq s$ implies $\mathcal{M} \models r \sqsubseteq s$ and every $t \in \Sigma$, $\mathcal{T} \models t \sqsubseteq r$ implies $\mathcal{M} \models t \sqsubseteq r$.

Additionally, a set \mathcal{M} is called an $\langle X, \Sigma \rangle$ -subsumption module of \mathcal{T} iff \mathcal{M} is an $\langle X, \Sigma \rangle$ -subsumee and $\langle X, \Sigma \rangle$ -subsumer module of \mathcal{T} . An $\langle X, \Sigma \rangle$ -subsumee (resp. subsumer, subsumption) justification is an $\langle X, \Sigma \rangle$ -subsumee (resp. subsumer, subsumption) module of \mathcal{T} that is minimal w.r.t. \subsetneq . Similarly, an $\langle r, \Sigma \rangle$ -role subsumption justification of \mathcal{T} is an $\langle r, \Sigma \rangle$ -role subsumption module of \mathcal{T} that is minimal w.r.t. \subsetneq .

We denote an $\langle X, \Sigma \rangle$ -subsumee (resp. subsumer, subsumption) justification as $\mathcal{J}_{\mathcal{T}}^{\leftarrow}(X, \Sigma)$ (resp. $\mathcal{J}_{\mathcal{T}}^{\rightarrow}(X, \Sigma)$, $\mathcal{J}_{\mathcal{T}}(X, \Sigma)$). Note that there may exist multiple $\langle X, \Sigma \rangle$ -(subsumer, subsumee) subsumption justifications. So we denote the set of all $\langle X, \Sigma \rangle$ -subsumee (resp. subsumer, subsumption) justifications as $\mathbb{J}_{\mathcal{T}}^{\leftarrow}(X, \Sigma)$ (resp. $\mathbb{J}_{\mathcal{T}}^{\rightarrow}(X, \Sigma)$, $\mathbb{J}_{\mathcal{T}}(X, \Sigma)$). Similarly, we denote a $\langle r, \Sigma \rangle$ -role subsumption justification of \mathcal{T} as $\mathcal{J}_{\mathcal{T}}(r, \Sigma)$ and the set of all $\langle r, \Sigma \rangle$ -role subsumption justifications of \mathcal{T} as $\mathbb{J}_{\mathcal{T}}(r, \Sigma)$.

According to the Definition 53, it is easy to compute subsumption justifications.

Theorem 54 Let \mathcal{T} be a normalised \mathcal{ELH} -terminology, let Σ be a signature, and let $X \in \mathbb{N}_{\mathbb{C}}$ and $r \in \mathbb{N}_{\mathbb{R}}$. Additionally, let $\mathbb{J}_{\mathcal{T}}^{\leftarrow}(X, \Sigma)$ (resp. $\mathbb{J}_{\mathcal{T}}^{\rightarrow}(X, \Sigma)$, $\mathbb{J}_{\mathcal{T}}(X, \Sigma)$) be the set of all $\langle X, \Sigma \rangle$ -subsumee justifications (resp. subsumer justification, subsumption justification). We have that

$$\mathbb{J}_{\mathcal{T}}^{\rightarrow}(X, \Sigma) := \text{Minimise}_{\subsetneq}(\mathbb{J}_{\mathcal{T}}^{\leftarrow}(X, \Sigma) \otimes \mathbb{J}_{\mathcal{T}}(X, \Sigma)).$$

Example 55 Let $\Sigma = \{A_1, A_2, B\}$ and $\mathcal{T} = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5\}$, where $\alpha_1 := X \equiv Y \sqcap Z$, $\alpha_2 := Y \sqsubseteq B$, $\alpha_3 := Z \equiv Z_1 \sqcap Z_2$, $\alpha_4 := A_1 \sqsubseteq Y$, and $\alpha_5 := A_2 \sqsubseteq Z \sqcap Z_1 \sqcap Z_2$. Then the sets $\mathcal{M}_1 = \{\alpha_1, \alpha_4, \alpha_5\}$, $\mathcal{M}_2 = \{\alpha_1, \alpha_3, \alpha_4, \alpha_5\}$, and \mathcal{T} are all $\langle X, \Sigma \rangle$ -subsumee modules of \mathcal{T} , whereas only \mathcal{M}_1 is an $\langle X, \Sigma \rangle$ -subsumee justification of \mathcal{T} . The set $\mathcal{M}_3 = \{\alpha_1, \alpha_2\}$ is an $\langle X, \Sigma \rangle$ -subsumer justification of \mathcal{T} . Finally, the set $\{\alpha_1, \alpha_2, \alpha_4, \alpha_5\}$ is an $\langle X, \Sigma \rangle$ -subsumption justification of \mathcal{T} .

Example 56 Let $\Sigma = \{r, t\}$ and let $\mathcal{T} = \{\alpha_1, \alpha_2\}$, where $\alpha_1 := r \sqsubseteq s$, $\alpha_2 := s \sqsubseteq t$. Then the set $\mathcal{M}_1 = \{\alpha_1, \alpha_2\}$ is an $\langle r, \Sigma \rangle$ -role subsumption justification of \mathcal{T} .

It is not difficult to get the following proposition from the primitive witnesses theorems 24 and Definition 53.

Proposition 57 Let \mathcal{T} be an \mathcal{ELH} -terminology, Σ be a signature, $X \in \Sigma \cap \mathbb{N}_{\mathbb{C}}$ and $r \in \Sigma \cap \mathbb{N}_{\mathbb{R}}$. Then the following statements hold:

- \mathcal{M} is an $\langle X, \Sigma \rangle$ -subsumer module of \mathcal{T} iff $X \notin \text{lhsWtn}_\Sigma(\mathcal{T}, \mathcal{M})$;
- \mathcal{M} is an $\langle X, \Sigma \rangle$ -subsumee module of \mathcal{T} iff $X \notin \text{rhsWtn}_\Sigma(\mathcal{T}, \mathcal{M})$;
- \mathcal{M} is a $\langle r, \Sigma \rangle$ -role subsumption module of \mathcal{T} iff $r \notin \text{roleWtn}_\Sigma(\mathcal{T}, \mathcal{M})$.

In the following subsections, we present algorithms for computing subsumer and subsumee justifications. The algorithms use the following notion of a cover of a set of sets.

Definition 58 (Cover) For a finite set S and a set $\mathbb{T} \subseteq 2^S$, we say that a set $\mathbb{M} \subseteq 2^S$ is a cover of \mathbb{T} iff $\mathbb{M} \subseteq \mathbb{T}$ and, for every $\mathcal{M} \in \mathbb{T}$, there exists $\mathcal{M}' \in \mathbb{M}$ such that $\mathcal{M}' \subseteq \mathcal{M}$.

In other words, a cover is a subset of \mathbb{T} containing all sets from \mathbb{T} that are minimal w.r.t. \subseteq . Therefore, a cover of the set of all subsumption modules also contains all subsumption justifications. We will use covers to characterise the output of our algorithms to ensure that all justifications have been computed.

3.1.1 Computing Role Subsumption Justifications

Algorithm 2 shows how to collect relevant Σ -role inclusions by its definition.

Algorithm 2: Computing the set of all Role Subsumption Justifications for Role Inclusions

```

1 function COVERR( $\mathcal{T}, \Sigma, r$ )
2    $\mathbb{M}_r = \{\emptyset\}$ 
3   for every  $s \in \Sigma \cap \mathbb{N}_R$  such that  $\mathcal{T} \models r \sqsubseteq s$  do
4      $\mathbb{M}_r := \mathbb{M}_r \otimes \text{Just}_{\mathcal{T}}(r \sqsubseteq s)$ 
5   end
6   for every  $t \in \Sigma \cap \mathbb{N}_R$  such that  $\mathcal{T} \models t \sqsubseteq r$  do
7      $\mathbb{M}_r := \mathbb{M}_r \otimes \text{Just}_{\mathcal{T}}(t \sqsubseteq r)$ 
8   end
9   return Minimise $\subseteq$ ( $\mathbb{M}_r$ )

```

It is easy to see that Algorithm 2 indeed computes all $\langle r, \Sigma \rangle$ -role subsumption justifications of \mathcal{T} w.r.t. Σ for $r \in \mathbb{N}_R$.

Proposition 59 Let \mathcal{T} be a normalised \mathcal{ELH} -terminology, Σ be a signature and $r \in \mathbb{N}_R$. Additionally, let $\mathbb{M} := \text{COVER}^R(\mathcal{T}, \Sigma, r)$ is computed by Algorithm 2. Then \mathbb{M} is the set of all $\langle r, \Sigma \rangle$ -role subsumption justifications.

3.1.2 Computing Subsumer Justifications

The algorithm for computing subsumer justifications relies on the notion of a subsumer simulation between terminologies, which is similar to the simulation notion between ontology hypergraphs introduced in [12, 34].

Definition 60 (Subsumer Simulation for \mathcal{ELH} -terminology) Let \mathcal{T}_1 and \mathcal{T}_2 be two normalised \mathcal{ELH} -terminologies and Σ a signature. A relation $S \subseteq \text{sig}^{\text{Nc}}(\mathcal{T}_1) \times \text{sig}^{\text{Nc}}(\mathcal{T}_2)$ is called a Σ -subsumer simulation from \mathcal{T}_1 to \mathcal{T}_2 iff the following conditions hold:

- ($S_{\text{Nc}}^{\rightarrow}$) if $(X_1, X_2) \in S$, then for every $B \in \Sigma$ with $\mathcal{T}_1 \models X_1 \sqsubseteq B$ it holds that $\mathcal{T}_2 \models X_2 \sqsubseteq B$; and
- ($S_{\exists}^{\rightarrow}$) if $(X_1, X_2) \in S$, then for each $Y_1 \bowtie_1 \exists r.Z_1 \in \mathcal{T}_1$ with $\mathcal{T}_1 \models X_1 \sqsubseteq Y_1$, $\mathcal{T}_1 \models r \sqsubseteq s$, $s \in \Sigma$, $\bowtie_1 \in \{\sqsubseteq, \equiv\}$, there exists $Y_2 \bowtie_2 \exists r'.Z_2 \in \mathcal{T}_2$ with $\mathcal{T}_2 \models X_2 \sqsubseteq Y_2$, $\bowtie_2 \in \{\sqsubseteq, \equiv\}$, $\mathcal{T}_2 \models r' \sqsubseteq s$, and $(Z_1, Z_2) \in S$.

We write $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$ iff there is a Σ -subsumer simulation S from \mathcal{T}_1 to \mathcal{T}_2 with $(X_1, X_2) \in S$. In particular, we write $\text{sim}_{\rightarrow}^{\mathcal{T}, \Sigma}(X_1, X_2)$ when $\mathcal{T}_1 = \mathcal{T}_2 = \mathcal{T}$.

A subsumer simulation conveniently captures the set of subsumers in the following sense: if a Σ -subsumer simulation from \mathcal{T}_1 to \mathcal{T}_2 contains the pair (X_1, X_2) , then X_2 entails w.r.t. \mathcal{T}_2 all subsumers of X_1 w.r.t. \mathcal{T}_1 that are formulated in the signature Σ . Formally, we obtain the following lemma.

Lemma 61 Let \mathcal{T}_1 and \mathcal{T}_2 be normalised \mathcal{ELH} -terminologies, and let Σ be a signature. If $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$, then, for every $D \in \mathcal{EL}_{\Sigma}$ with $\mathcal{T}_1 \models X_1 \sqsubseteq D$, it holds that $\mathcal{T}_2 \models X_2 \sqsubseteq D$.

Algorithm 3: Computing the set of all Subsumer Justifications for an \mathcal{ELH} -terminology

```

1 function COVER $_{\rightarrow}$  ( $\mathcal{T}, X, \Sigma$ )
2    $\mathbb{M}_X^{\rightarrow} = \{\emptyset\}$ 
3   for every  $B \in \Sigma \cap \text{Nc}$  such that  $\mathcal{T} \models X \sqsubseteq B$  do
4      $\mathbb{M}_X^{\rightarrow} := \mathbb{M}_X^{\rightarrow} \otimes \text{Just}_{\mathcal{T}}(X \sqsubseteq B)$ 
5   end
6   for every  $Y \bowtie_1 \exists r.Z \in \mathcal{T}$  ( $\bowtie_1 \in \{\sqsubseteq, \equiv\}$ ) and  $s \in \Sigma \cap \text{Nr}$  such that  $\mathcal{T} \models X \sqsubseteq Y$  and
    $\mathcal{T} \models r \sqsubseteq s$  do
7      $\mathbb{M}_{\exists s.Z}^{\rightarrow} := \emptyset$ 
8     for every  $Y' \bowtie_2 \exists r'.Z' \in \mathcal{T}$  ( $\bowtie_2 \in \{\sqsubseteq, \equiv\}$ ) such that  $\mathcal{T} \models X \sqsubseteq Y'$ ,  $\mathcal{T} \models r' \sqsubseteq s$ ,
        $\text{sim}_{\rightarrow}^{\mathcal{T}, \Sigma}(Z, Z')$  do
9        $\mathbb{M}_{Z'}^{\rightarrow} := \text{COVER}_{\rightarrow}(\mathcal{T}, Z', \Sigma)$ 
10       $\mathbb{M}_{\exists s.Z}^{\rightarrow} := \mathbb{M}_{\exists s.Z}^{\rightarrow} \cup (\{\{Y' \bowtie_2 \exists r'.Z'\}\} \otimes \mathbb{M}_{Z'}^{\rightarrow} \otimes \text{Just}_{\mathcal{T}}(X \sqsubseteq Y') \otimes \text{Just}_{\mathcal{T}}(r' \sqsubseteq s))$ 
11    end
12     $\mathbb{M}_X^{\rightarrow} := \mathbb{M}_X^{\rightarrow} \otimes \mathbb{M}_{\exists s.Z}^{\rightarrow}$ 
13  end
14  return  $\text{Minimise}_{\sqsubseteq}(\mathbb{M}_X^{\rightarrow})$ 

```

Guided by the subsumer simulation notion, we can devise our algorithm for computing subsumer justifications. Algorithm 3 computes the subsumer justifications for an acyclic normalised \mathcal{ELH} -terminology \mathcal{T} , a signature Σ , and a concept name X . Lines 3–13 of the algorithm compute all $\langle X, \Sigma \rangle$ -subsumption modules of \mathcal{T} . To ensure that the returned modules are minimal w.r.t. \sqsubseteq , the

algorithm calls the function $\text{Minimise}_{\subseteq}(\mathbb{M}_{\bar{X}}^{\rightarrow})$ in Line 14, which removes any set in $\mathbb{M}_{\bar{X}}^{\rightarrow}$ that is not minimal.

We illustrate Algorithm 3 with the following two examples. First example, let $\mathcal{T} = \{X \sqsubseteq B, X \sqsubseteq Y, Y \sqsubseteq B\}$ and $\Sigma = \{B\}$. Consider the execution of $\text{COVER}_{\rightarrow}(\mathcal{T}, X, \Sigma)$. In Line 4, $\mathbb{M}_{\bar{X}}^{\rightarrow}$ is set to $\text{Just}_{\mathcal{T}}(X \sqsubseteq B)$, where $\text{Just}_{\mathcal{T}}(X \sqsubseteq B) = \{\{X \sqsubseteq B\}, \{X \sqsubseteq Y, Y \sqsubseteq B\}\}$. Since there are no axioms of the form $Y \sqsubseteq \exists r.Z \in \mathcal{T}$ or $Y \equiv \exists r.Z \in \mathcal{T}$, the lines 6–14 have no effect. Finally, the algorithm returns $\mathbb{M}_{\bar{X}}^{\rightarrow}$ in Line 14.

For the second example, let $\mathcal{T} = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5\}$ and $\Sigma = \{A, B, s\}$, where $\alpha_1 = X \sqsubseteq \exists r.A$, $\alpha_2 = X \sqsubseteq \exists r.B$, $\alpha_3 = X \sqsubseteq \exists r.Y$, $\alpha_4 = Y \equiv A \sqcap B$, and $\alpha_5 = r \sqsubseteq s$. We consider again the execution of $\text{COVER}_{\rightarrow}(\mathcal{T}, X, \Sigma)$. We proceed to Line 5 as there are no concept names in Σ entailed by X w.r.t. \mathcal{T} . However, the concepts $\exists r.A$, $\exists r.B$ and $\exists r.Y$ are entailed by X w.r.t. \mathcal{T} . It holds that $\text{sim}_{\rightarrow, \Sigma}^{\mathcal{T}}(Z, Z')$ for every $(Z, Z') \in \{(A, A), (B, B), (Y, Y), (A, Y), (B, Y)\}$, whereas $\text{sim}_{\rightarrow, \Sigma}^{\mathcal{T}}(Z, Z')$ does not hold for any $(Z, Z') \in \{(A, B), (B, A), (Y, A), (Y, B)\}$. Therefore, for every $Z \in \{A, B, Y\}$ the recursive call $\text{COVER}_{\rightarrow}(\mathcal{T}, Z, \Sigma)$ is made in Line 9. The following sets are computed in lines 6–12: $\mathbb{M}_A^{\rightarrow} = \{\emptyset\}$, $\mathbb{M}_B^{\rightarrow} = \{\emptyset\}$, and $\mathbb{M}_Y^{\rightarrow} = \{\{\alpha_4\}\}$ as well as

$$\begin{aligned} \mathbb{M}_{\exists s.A}^{\rightarrow} &= (\{\alpha_1, \alpha_5\} \otimes \mathbb{M}_A^{\rightarrow}) \cup (\{\alpha_3, \alpha_5\} \otimes \mathbb{M}_Y^{\rightarrow}) = \{\{\alpha_1, \alpha_5\}, \{\alpha_3, \alpha_4, \alpha_5\}\} \\ \mathbb{M}_{\exists s.B}^{\rightarrow} &= (\{\alpha_2, \alpha_5\} \otimes \mathbb{M}_B^{\rightarrow}) \cup (\{\alpha_3, \alpha_5\} \otimes \mathbb{M}_Y^{\rightarrow}) = \{\{\alpha_2, \alpha_5\}, \{\alpha_3, \alpha_4, \alpha_5\}\} \\ \mathbb{M}_{\exists s.Y}^{\rightarrow} &= \{\alpha_3, \alpha_5\} \otimes \mathbb{M}_Y^{\rightarrow} = \{\{\alpha_3, \alpha_4, \alpha_5\}\} \\ \text{Minimise}_{\subseteq}(\mathbb{M}_{\bar{X}}^{\rightarrow}) &= \text{Minimise}_{\subseteq}(\mathbb{M}_{\exists s.A}^{\rightarrow} \otimes \mathbb{M}_{\exists s.B}^{\rightarrow} \otimes \mathbb{M}_{\exists s.Y}^{\rightarrow}) = \{\{\alpha_3, \alpha_4, \alpha_5\}\}. \end{aligned}$$

Finally, $\text{COVER}_{\rightarrow}(\mathcal{T}, X, \Sigma)$ returns $\text{Minimise}_{\subseteq}(\mathbb{M}_{\bar{X}}^{\rightarrow}) = \{\{\alpha_3, \alpha_4, \alpha_5\}\}$ in Line 14.

The following theorem shows that Algorithm 3 indeed computes the set of subsumer modules, thus producing a cover of subsumer justifications.

Theorem 62 *Let \mathcal{T} be an acyclic, normalised \mathcal{ELH} -terminology, and let Σ be a signature. Let $X \in \mathbf{N}_{\mathcal{C}}$, and let $\mathbb{M}_{\bar{X}}^{\rightarrow} := \text{COVER}_{\rightarrow}(\mathcal{T}, X, \Sigma)$ computed by Algorithm 3. Then $\mathbb{M}_{\bar{X}}^{\rightarrow}$ is the set of all $\langle X, \Sigma \rangle$ -subsumer justifications of \mathcal{T} .*

Observe that $\text{COVER}_{\rightarrow}(\mathcal{T}, X, \Sigma)$ may be called several times during the execution of Algorithm 3. The algorithm can be optimised by caching the return value of the first execution, and retrieving it from memory for subsequent calls.

3.1.3 Computing Subsumee Justifications

Similar with the algorithm for computing subsumer justifications, algorithm for computing subsumee justifications relies on the notion of subsumee simulation between terminologies, which we introduce below. Before that, we first introduce the notion of Σ -entailment.

Definition 63 (Σ -Entailment for \mathcal{ELH} -terminology) Let Σ be a signature and \mathcal{T} a normalised \mathcal{ELH} -terminology. We say that $X \in \mathbf{N}_C$ is Σ -entailed w.r.t. \mathcal{T} iff there exists $C \in \mathcal{EL}_\Sigma$ with $\mathcal{T} \models C \sqsubseteq X$; and $r \in \mathbf{N}_R$ is Σ -entailed w.r.t. \mathcal{T} iff there exists $s \in \Sigma \cap \mathbf{N}_R$ with $\mathcal{T} \models s \sqsubseteq r$.

Example 64 Let $\mathcal{T} = \{X \equiv X_1 \sqcap X_2, B_1 \sqsubseteq X_1, X_2 \equiv \exists r.Z, B_2 \sqsubseteq Z, s \sqsubseteq r\}$. We have that

$$\text{non-conj}_{\mathcal{T}}(X) = \{X_1, X_2\}.$$

Then r is Σ -entailed w.r.t. \mathcal{T} ; X is complex Σ -entailed w.r.t. \mathcal{T} for $\Sigma = \{B_1, B_2, s\}$; but X is not complex Σ' -entailed w.r.t. \mathcal{T} , where Σ' ranges over $\{B_1, B_2\}, \{B_1, s\}, \{B_2, s\}$. Additionally, X is not complex Σ -entailed w.r.t. $\mathcal{T} \cup \{B_1 \sqsubseteq X\}$.

Definition 65 (Subsumee Simulation for \mathcal{ELH} -terminology) Let $\mathcal{T}_1, \mathcal{T}_2$ be normalised \mathcal{ELH} -terminologies, and let Σ be a signature. A relation $S \subseteq \text{sig}^{\mathbf{N}_C}(\mathcal{T}_1) \times \text{sig}^{\mathbf{N}_C}(\mathcal{T}_2)$ is a Σ -subsumee simulation from \mathcal{T}_1 to \mathcal{T}_2 iff the following conditions are satisfied:

($S_{\mathbf{N}_C}^{\leftarrow}$) if $(X_1, X_2) \in S$, then for every $B \in \Sigma$ with $\mathcal{T}_1 \models B \sqsubseteq X_1$, it holds that $\mathcal{T}_2 \models B \sqsubseteq X_2$;

(S_{\exists}^{\leftarrow}) if $(X_1, X_2) \in S$, $X_1 \equiv \exists r.Y_1 \in \mathcal{T}_1$ and $\mathcal{T}_1 \models s \sqsubseteq r$, $s \in \Sigma$ and Y_1 is Σ -entailed w.r.t. \mathcal{T}_1 , then for every $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$, there exists $X'_2 \equiv \exists r'.Y_2 \in \mathcal{T}_2$ such that $\mathcal{T}_2 \models s \sqsubseteq r'$ and $(Y_1, Y_2) \in S$;

(S_{\sqcap}^{\leftarrow}) if $(X_1, X_2) \in S$ and $X_1 \equiv Y_1 \sqcap \dots \sqcap Y_n \in \mathcal{T}_1$, then for every $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$, there exists $X'_1 \in \text{non-conj}_{\mathcal{T}_1}(X_1)$ with $(X'_1, X'_2) \in S$.

We write $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\leftarrow} \langle \mathcal{T}_2, X_2 \rangle$ iff there exists a Σ -subsumee simulation S from \mathcal{T}_1 to \mathcal{T}_2 with $(X_1, X_2) \in S$.

In Line 6 of Algorithm 4, the definition of *complex Σ -entailment* is given as follows.

Definition 66 (Complex Σ -Entailment) Let \mathcal{T} be a normalized \mathcal{ELH} -terminology, Σ be a signature, and let $X \in \mathbf{N}_C$ and $r \in \mathbf{N}_R$.

We say that X is complex Σ -entailed w.r.t. \mathcal{T} iff for every $Y \in \text{non-conj}_{\mathcal{T}}(X)$ at least one of the following conditions holds:

- there exists $B \in \Sigma$ such that $\mathcal{T} \models B \sqsubseteq Y$ and $\mathcal{T} \not\models B \sqsubseteq X$;
- there exists $Y \equiv \exists r.Z \in \mathcal{T}$ and r and Z are each Σ -entailed in \mathcal{T} .

Otherwise, X is said to be simply Σ -entailed.

Analogously to subsumer simulations, a subsumee simulation captures the set of subsumees as it is made precise in the following lemma.

Lemma 67 Let \mathcal{T}_1 and \mathcal{T}_2 be two normalised \mathcal{ELH} -terminologies, and let Σ be a signature. If $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\leftarrow} \langle \mathcal{T}_2, X_2 \rangle$, then $\mathcal{T}_2 \models D \sqsubseteq X_2$ holds for every $D \in \mathcal{EL}_\Sigma$ with $\mathcal{T}_1 \models D \sqsubseteq X_1$.

Now we present some auxiliary notions for handling conjunctions on the left-hand side of subsumptions.

We define for each concept name X a so-called *definitorial forest* consisting of sets of axioms of the form $Y \equiv Y_1 \sqcap \dots \sqcap Y_n$ which can be thought of as forming *trees*. Any $\langle X, \Sigma \rangle$ -subsumee justification contains the axioms of a selection of these trees, i.e., one tree for every conjunction formulated over Σ that entails X w.r.t. \mathcal{T} . Formally, we define $\text{DefForest}_{\mathcal{T}}^{\sqcap}(X) \subseteq 2^{\mathcal{T}}$ to be the smallest set closed under the following conditions:

- $\emptyset \in \text{DefForest}_{\mathcal{T}}^{\sqcap}(X)$;
- $\{\alpha\} \in \text{DefForest}_{\mathcal{T}}^{\sqcap}(X)$ for $\alpha := X \equiv X_1 \sqcap \dots \sqcap X_n \in \mathcal{T}$; and
- $\Gamma \cup \{\alpha\} \in \text{DefForest}_{\mathcal{T}}^{\sqcap}(X)$ for $\Gamma \in \text{DefForest}_{\mathcal{T}}^{\sqcap}(X)$ with $Z \equiv Z_1 \sqcap \dots \sqcap Z_k \in \Gamma$ and $\alpha := Z_i \equiv Z_i^1 \sqcap \dots \sqcap Z_i^n \in \mathcal{T}$.

Given $\Gamma \in \text{DefForest}_{\mathcal{T}}^{\sqcap}(X)$, we set $\text{leaves}(\Gamma) := \text{sig}(\Gamma) \setminus \{X \in \text{sig}(C) \mid X \equiv C \in \Gamma\}$ if $\Gamma \neq \emptyset$; and $\{X\}$ otherwise. We denote the maximal element of $\text{DefForest}_{\mathcal{T}}^{\sqcap}(X)$ w.r.t. \subseteq with $\text{max-tree}_{\mathcal{T}}^{\sqcap}(X)$. Finally, we set $\text{non-conj}_{\mathcal{T}}(X) := \text{leaves}(\text{max-tree}_{\mathcal{T}}^{\sqcap}(X))$.

Example 68 Let $\mathcal{T} = \{\alpha_1, \alpha_2, \alpha_3\}$, where $\alpha_1 := X \equiv Y \sqcap Z$, $\alpha_2 := Y \equiv Y_1 \sqcap Y_2$, and $\alpha_3 := Z \equiv Z_1 \sqcap Z_2$. Then $\text{DefForest}_{\mathcal{T}}^{\sqcap}(X) = \{\emptyset, \{\alpha_1\}, \{\alpha_1, \alpha_2\}, \{\alpha_1, \alpha_3\}, \{\alpha_1, \alpha_2, \alpha_3\}\}$. We have that $\text{leaves}(\{\alpha_1, \alpha_3\}) = \{Y, Z_1, Z_2\}$, $\text{max-tree}_{\mathcal{T}}^{\sqcap}(X) = \{\alpha_1, \alpha_2, \alpha_3\}$, and $\text{non-conj}_{\mathcal{T}}(X) = \{Y_1, Y_2, Z_1, Z_2\}$.

We definite definitorial forest in order to enumerate all possible situations and find all possible sub-trees such that $\text{Case}(S_{\exists}^{\leftarrow, \mathcal{L}})$ can be satisfied.

Using the notion of a subsumee simulation, we can device Algorithm 4 for computing a cover of the subsumee justifications for a given normalized \mathcal{ELH} -terminology \mathcal{T} , a concept name X , and a signature Σ . The function for obtaining the $\langle X, \Sigma \rangle$ -subsumee justifications of \mathcal{T} is $\text{COVER}_{\leftarrow}(\mathcal{T}, X, \Sigma, \mathcal{T}, X)$. Note that Algorithm 5, Algorithm 6, and Algorithm 7 are called as subroutines in Line 5, 10 and 13 in Algorithm 4. The five different parameters for Algorithm 4 are needed due to the recursive calls in Algorithm 7 (Line 11) and Algorithm 6 (Line 9).

The existence of axiom $\alpha_{X_1} := X_1 \equiv \exists r. Y_1 \in \mathcal{T}_1$ in Line 2 of Algorithm 6 is guaranteed by Line 9 of Algorithm 4. The axiom $\alpha_{X'_2} := X'_2 \equiv \exists r'. Y'_2 \in \mathcal{T}_2$ in Line 8 of Algorithm 6 exists as we assume that $\langle \mathcal{T}_2, X_2 \rangle$ subsumee-simulates $\langle \mathcal{T}_1, X_1 \rangle$. Moreover, there is at most one axiom $\alpha_{X_1} \in \mathcal{T}_1$ and at most one $\alpha_{X'_2} \in \mathcal{T}_2$ as \mathcal{T}_1 and \mathcal{T}_2 are terminologies. The concept name X_2 may be defined as a conjunction in \mathcal{T}_2 whose conjuncts in turn may also be defined as a conjunction in \mathcal{T}_2 and so forth. In Line 3 of Algorithm 6 all axioms forming the maximal resulting definitorial conjunctive tree are collected.

For Algorithm 7, we define

$$\text{def}_{\mathcal{T}}^{\sqcap} := \{X \in \text{sig}^{\text{Nc}}(\mathcal{T}) \mid X \equiv Y_1 \sqcap \dots \sqcap Y_n \in \mathcal{T}\}$$

to be the set of concept names that are conjunctively defined in \mathcal{T} . For every $X \in \text{def}_{\mathcal{T}}^{\square}$, we set $\text{def}_{\mathcal{T}}^{\square}(X) := \alpha$, where $\alpha = X \equiv Y_1 \sqcap \dots \sqcap Y_n \in \mathcal{T}$.

The axiom $\alpha_{X_1} := X_1 \equiv Y_1 \sqcap \dots \sqcap Y_m \in \mathcal{T}_1$ in Line 2 of Algorithm 7 is guaranteed by Line 12 of Algorithm 4. In case X_2 is defined as a conjunction in \mathcal{T}_2 , the pair consisting of \mathcal{T}_2 containing only a partial conjunctive tree rooted at X_2 and X_2 needs to be considered to subsume simulate X_1 in \mathcal{T}_1 . Therefore Algorithm 7 considers every partial conjunctive tree Γ from $\text{DefForest}_{\mathcal{T}_2}^{\square}(X_2)$ in Line 4 and removes the axioms in δ_{Γ} connecting the leaves of Γ with the remaining conjunctive tree from \mathcal{T}_2 in lines 10 and 11.

Algorithm 4: Computing the set of all Subsumee Justifications for an \mathcal{ELH} -terminology

```

1 function COVER $_{\leftarrow}$ ( $\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2$ )
2   if  $X_1$  is not  $\Sigma$ -entailed w.r.t.  $\mathcal{T}_1$  then
3     | return  $\{\emptyset\}$ 
4   end
5    $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \text{COVER}_{\leftarrow}^{\text{Nc}}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2)$ 
6   if  $X_1$  is not complex  $\Sigma$ -entailed in  $\mathcal{T}_1$  then
7     | return  $\text{Minimise}_{\subseteq}(\mathbb{M}_{(X_1, X_2)}^{\leftarrow})$ 
8   end
9   if  $X_1 \equiv \exists r.Y \in \mathcal{T}_1$ , and  $r, Y$  are  $\Sigma$ -entailed w.r.t.  $\mathcal{T}_1$  then
10    |  $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \mathbb{M}_{(X_1, X_2)}^{\leftarrow} \otimes \text{COVER}_{\leftarrow}^{\exists}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2)$ 
11  end
12  else if  $X_1 \equiv Y_1 \sqcap \dots \sqcap Y_m \in \mathcal{T}_1$  then
13    |  $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \mathbb{M}_{(X_1, X_2)}^{\leftarrow} \otimes \text{COVER}_{\leftarrow}^{\square}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2)$ 
14  end
15  return  $\text{Minimise}_{\subseteq}(\mathbb{M}_{(X_1, X_2)}^{\leftarrow})$ 

```

Algorithm 5: Computing the set of all Subsumee Justifications for an \mathcal{ELH} -terminology – Local Case

```

1 function COVER $_{\leftarrow}^{\text{Nc}}$  ( $\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2$ )
2    $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} = \{\emptyset\}$ 
3   for every  $B \in \Sigma \cap \text{Nc}$  such that  $\mathcal{T}_1 \models B \sqsubseteq X_1$  do
4     |  $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \mathbb{M}_{(X_1, X_2)}^{\leftarrow} \otimes \text{Just}_{\mathcal{T}_2}(B \sqsubseteq X_2)$ 
5   end
6   return  $\text{Minimise}_{\subseteq}(\mathbb{M}_{(X_1, X_2)}^{\leftarrow})$ 

```

We illustrate Algorithm 4 with the following example.

Example 69 Let $\mathcal{T} = \{X \equiv \exists r.Y, Y \equiv \exists s.Z, Z \equiv A \sqcap Z', A \sqsubseteq B, B \sqsubseteq Z', Z' \sqsubseteq A\}$ be an \mathcal{ELH} -terminology, and let $\Sigma = \{A, B, r, s\}$ be a signature. It can readily be verified that \mathcal{T} is normalised.

Algorithm 6: Computing the set of all Subsumee Justifications for an \mathcal{ELH} -terminology – Existential Case

```

1 function COVER $_{\leftarrow}^{\exists}$  ( $\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2$ )
2   let  $\alpha_{X_1} := X_1 \equiv \exists r.Y_1 \in \mathcal{T}_1$ 
3    $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \{\text{max-tree}_{\mathcal{T}_2}^{\square}(X_2)\}$ 
4   for every  $s \in \Sigma \cap N_R$  such that  $\mathcal{T}_1 \models s \sqsubseteq r$  do
5     for every  $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$  do
6       let  $\alpha_{X'_2} := X'_2 \equiv \exists r'.Y'_2 \in \mathcal{T}_2$  such that  $\mathcal{T}_2 \models s \sqsubseteq r'$ 
7        $\mathbb{M}_{Y'_2}^{\leftarrow} := \text{COVER}_{\leftarrow}(\mathcal{T}_1, Y_1, \Sigma, \mathcal{T}_2, Y'_2)$ 
8        $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \mathbb{M}_{(X_1, X_2)}^{\leftarrow} \otimes \{\{\alpha_{X'_2}\}\} \otimes \text{Just}_{\mathcal{T}_2}(s \sqsubseteq r') \otimes \mathbb{M}_{Y'_2}^{\leftarrow}$ 
9     end
10  end
11  return Minimise $_{\subseteq}(\mathbb{M}_{(X_1, X_2)}^{\leftarrow})$ 

```

To determine the $\langle X, \Sigma \rangle$ -subsumee justifications of \mathcal{T} , we need to execute Algorithm 4 computing $\text{COVER}_{\leftarrow}(\mathcal{T}, X, \Sigma, \mathcal{T}, X)$. As X is (complex) Σ -entailed, Lines 2–4 and Lines 6–8 are omitted. In Line 5, $\text{COVER}_{\leftarrow}^{\text{Nc}}(\mathcal{T}, X, \Sigma, \mathcal{T}, X)$ is called which is implemented in Algorithm 4.

The for-loop in Lines 3–5 of Algorithm 5 does not apply as $\mathcal{T} \not\models A \sqsubseteq X$ and $\mathcal{T} \not\models B \sqsubseteq X$. We obtain $\text{COVER}_{\leftarrow}^{\text{Nc}}(\mathcal{T}, X, \Sigma, \mathcal{T}, X) = \{\emptyset\}$, and then backtrack to Line 5 of Algorithm 4. The if-statement in Line 9 applies as \mathcal{T} contains an axiom of the form $X \equiv \exists r.Y$ and both, X and r , are Σ -entailed w.r.t. \mathcal{T} . We proceed in Line 10 with calling $\text{COVER}_{\leftarrow}^{\exists}(\mathcal{T}, X, \Sigma, \mathcal{T}, X)$, which is implemented in Algorithm 6.

We obtain $\mathbb{M}_{(X, X)}^{\leftarrow} := \{\text{max-tree}_{\mathcal{T}}^{\square}(X)\} = \{\emptyset\}$ in Line 3 of Algorithm 6. The for-loop from Line 4 to 10 is executed for $r \in \Sigma$ as $\mathcal{T} \models r \sqsubseteq r$. The nested for-loop in Lines 5–9 is executed for X as $\text{non-conj}_{\mathcal{T}}(X) = \{X\}$. In Line 6 $\alpha_{X'_2}$ is set to $X \equiv \exists r.Y$ and, in Line 8, the recursive call $\text{COVER}_{\leftarrow}(\mathcal{T}, Y, \Sigma, \mathcal{T}, Y)$ is made, which is computed in Algorithm 4.

Then, in Line 10 of Algorithm 4, $\text{COVER}_{\leftarrow}^{\exists}(\mathcal{T}, Y, \Sigma, \mathcal{T}, Y)$ is called as Y is complex Σ -entailed w.r.t. \mathcal{T} , $Y \equiv \exists s.Z \in \mathcal{T}$, and s, Z are each Σ -entailed w.r.t. \mathcal{T} . Similar to $\text{COVER}_{\leftarrow}^{\exists}(\mathcal{T}, X, \Sigma, \mathcal{T}, X)$, the execution of $\text{COVER}_{\leftarrow}^{\exists}(\mathcal{T}, Y, \Sigma, \mathcal{T}, Y)$ causes another recursive call $\text{COVER}_{\leftarrow}(\mathcal{T}, Z, \Sigma, \mathcal{T}, Z)$ from Line 7 of Algorithm 6.

As Z is Σ -entailed w.r.t. \mathcal{T} , we have that $\text{COVER}_{\leftarrow}^{\text{Nc}}(\mathcal{T}, Z, \Sigma, \mathcal{T}, Z)$ is executed. The for-loop starting in Line 3 of Algorithm 5 applies as $\mathcal{T} \models A \sqsubseteq Z$ and $\mathcal{T} \models B \sqsubseteq Z$. We obtain $\mathbb{M}_Z^{\leftarrow} := \text{Just}_{\mathcal{T}}(A \sqsubseteq Z) \otimes \text{Just}_{\mathcal{T}}(B \sqsubseteq Z)$, where $\text{Just}_{\mathcal{T}}(A \sqsubseteq Z) = \text{Just}_{\mathcal{T}}(B \sqsubseteq Z) = \{Z \equiv A \cap Z', A \sqsubseteq B, B \sqsubseteq Z', Z' \sqsubseteq A\}$. This finishes the call $\text{COVER}_{\leftarrow}^{\text{Nc}}(\mathcal{T}, Z, \Sigma, \mathcal{T}, Z)$, and we backtrack to Line 4 of $\text{COVER}_{\leftarrow}(\mathcal{T}, Z, \Sigma, \mathcal{T}, Z)$. As Z is not complex Σ -entailed, this finishes the call $\text{COVER}_{\leftarrow}(\mathcal{T}, Z, \Sigma, \mathcal{T}, Z)$ with $\mathbb{M}_Z^{\leftarrow} = \{Z \equiv A \cap Z', A \sqsubseteq B, B \sqsubseteq Z', Z' \sqsubseteq A\}$.

We backtrack to Line 10 of $\text{COVER}_{\leftarrow}^{\exists}(\mathcal{T}, Y, \Sigma, \mathcal{T}, Y)$ and set $\mathbb{M}_Y^{\leftarrow} := \mathbb{M}_Y^{\leftarrow} \otimes \{\{Y \equiv \exists s.Z\}\} \otimes \mathbb{M}_Z^{\leftarrow}$ which yields $\mathbb{M}_Y^{\leftarrow} = \{Y \equiv \exists s.Z, Z \equiv A \cap Z', A \sqsubseteq B, B \sqsubseteq Z', Z' \sqsubseteq A\}$. This finishes the call

Algorithm 7: Computing the set of all Subsumee Justifications for an \mathcal{ELH} -terminology – Conjunctive Case

```

1 function COVER $_{\leftarrow}^{\square}$ ( $\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2$ )
2   let  $\alpha_{X_1} := X_1 \equiv Y_1 \sqcap \dots \sqcap Y_m \in \mathcal{T}_1$ 
3    $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \emptyset$ 
4   for  $\Gamma \in \text{DefForest}_{\mathcal{T}_2}^{\square}(X_2)$  do
5     Let  $\delta_{\Gamma} := \{ \text{def}_{\mathcal{T}_2}^{\square}(X') \mid X' \in \text{leaves}(\Gamma) \cap \text{def}_{\mathcal{T}_2}^{\square} \}$ 
6      $\mathbb{M}_{\Gamma}^{\leftarrow} := \{ \Gamma \}$ 
7     for  $X'_2 \in \text{leaves}(\Gamma)$  do
8        $\mathbb{M}_{X'_2}^{\leftarrow} := \emptyset$ 
9       for every  $X'_1 \in \text{non-conj}_{\mathcal{T}_1}(X_1)$  do
10        if  $\langle \mathcal{T}_1, X'_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\leftarrow} \langle \mathcal{T}_2 \setminus \delta_{\Gamma}, X'_2 \rangle$  then
11           $\mathbb{M}_{X'_2}^{\leftarrow} := \mathbb{M}_{X'_2}^{\leftarrow} \cup \text{COVER}_{\leftarrow}(\mathcal{T}_1, X'_1, \Sigma, \mathcal{T}_2 \setminus \delta_{\Gamma}, X'_2)$ 
12        end
13      end
14       $\mathbb{M}_{\Gamma}^{\leftarrow} := \mathbb{M}_{\Gamma}^{\leftarrow} \otimes \mathbb{M}_{X'_2}^{\leftarrow}$ 
15    end
16     $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \mathbb{M}_{(X_1, X_2)}^{\leftarrow} \cup \mathbb{M}_{\Gamma}^{\leftarrow}$ 
17  end
18  return  $\text{Minimise}_{\subseteq}(\mathbb{M}_{(X_1, X_2)}^{\leftarrow})$ 

```

$\text{COVER}_{\leftarrow}^{\exists}(\mathcal{T}, Y, \Sigma, \mathcal{T}, Y)$ and it backtracks to Line 9 and ends the call $\text{COVER}_{\leftarrow}(\mathcal{T}, Y, \Sigma, \mathcal{T}, Y)$. We set $\mathbb{M}_X^{\leftarrow} := \mathbb{M}_X^{\leftarrow} \otimes \{ \{ X \equiv \exists r.Y \} \} \otimes \mathbb{M}_Y^{\leftarrow}$ in Line 8 of Algorithm 6 for $\text{COVER}_{\leftarrow}^{\exists}(\mathcal{T}, X, \Sigma, \mathcal{T}, X)$. Thus $\text{COVER}_{\leftarrow}^{\exists}(\mathcal{T}, X, \Sigma, \mathcal{T}, X)$ returns $\mathbb{M}_X^{\leftarrow} = \{ \{ X \equiv \exists r.Y, Y \equiv \exists s.Z, Z \equiv A \sqcap Z', A \sqsubseteq B, B \sqsubseteq Z', Z' \sqsubseteq A \} \}$ and we backtrack to Line 10 of Algorithm 4. Finally, all sets that are not minimal w.r.t. \subseteq are removed from $\mathbb{M}_X^{\leftarrow}$ in Line 15, which ends the execution of $\text{COVER}_{\leftarrow}(\mathcal{T}, X, \Sigma, \mathcal{T}, X)$.

The following theorem shows that Algorithm 4 indeed computes a cover of the set of subsumee modules, which are all subsumee justifications after the function $\text{Minimise}_{\subseteq}$.

Theorem 70 *Let \mathcal{T} be an acyclic, normalised \mathcal{ELH} -terminology, $X \in \mathbb{N}_{\mathcal{C}}$ and Σ be a signature. Additionally, let $\mathbb{M}_X^{\leftarrow} := \text{COVER}_{\leftarrow}(\mathcal{T}, X, \Sigma, \mathcal{T}, X)$ computed by Algorithm 4. Then $\mathbb{M}_X^{\leftarrow}$ is the set of all $\langle X, \Sigma \rangle$ -subsumee justifications of \mathcal{T} .*

The number of subsumption (subsumer, subsumee) justifications depends on \mathcal{T} and Σ . Clearly, it can be exponential many in the size of the terminology. Note that the simulation checks can be performed in polynomial time [12, 34]. Therefore, our algorithm for computing subsumption (subsumer, subsumee) justifications runs in exponential time w.r.t. the size of input terminology in the worst case.

As we can see from the algorithms given in this section, the reason that there might exist exponential many subsumption justifications is caused by the fact that there can be exponential many justifications w.r.t. an inclusion α in the worst case. However, it is obvious that if the number of justifications w.r.t. an inclusion α is bounded by a constant number, which happens often in real world ontologies, the number of subsumption justifications is also bounded. In this case, our algorithm for subsumer justifications becomes polynomial, but the algorithm for subsumee and subsumption justifications are still exponential in the worst case.

Proposition 71 *Let \mathcal{T} be a normalised \mathcal{ELH} -terminology, let Σ be a signature and $X \in \Sigma$. The algorithm of computing all $\langle X, \Sigma \rangle$ -subsumption justifications runs in EXPTIME.*

3.1.4 Proofs

Proofs about \otimes

Lemma 72 *Let S be a set. Additionally, let $\mathbb{T}_1, \mathbb{T}_2 \subseteq 2^S$ and let $\mathbb{M}_1, \mathbb{M}_2 \subseteq 2^S$ such that \mathbb{M}_i is a cover of \mathbb{T}_i for $i \in \{1, 2\}$. Then $\mathbb{M}_1 \otimes \mathbb{M}_2$ is a cover of $\mathbb{T}_1 \otimes \mathbb{T}_2$.*

Proof We have to show that:

- (i) $\mathbb{M}_1 \otimes \mathbb{M}_2 \subseteq \mathbb{T}_1 \otimes \mathbb{T}_2$; and
- (ii) for every $\mathcal{M} \in \mathbb{T}_1 \otimes \mathbb{T}_2$, there exists $\mathcal{M}' \in \mathbb{M}_1 \otimes \mathbb{M}_2$ such that $\mathcal{M}' \subseteq \mathcal{M}$.

For (i), let $\mathcal{M} \in \mathbb{M}_1 \otimes \mathbb{M}_2$, i.e. $\mathcal{M} = \mathcal{M}_1 \cup \mathcal{M}_2$ with $\mathcal{M}_1 \in \mathbb{M}_1$ and $\mathcal{M}_2 \in \mathbb{M}_2$. As \mathbb{M}_i is a cover of \mathbb{T}_i , it holds that $\mathcal{M}_i \in \mathbb{T}_i$ for $i \in \{1, 2\}$. Consequently, we have $\mathcal{M} = \mathcal{M}_1 \cup \mathcal{M}_2 \in \mathbb{T}_1 \otimes \mathbb{T}_2$.

For (ii), let $\mathcal{M} \in \mathbb{T}_1 \otimes \mathbb{T}_2$, i.e. $\mathcal{M} = \mathcal{M}_1 \cup \mathcal{M}_2$ with $\mathcal{M}_1 \in \mathbb{T}_1$ and $\mathcal{M}_2 \in \mathbb{T}_2$. Hence, there exists $\mathcal{M}'_i \in \mathbb{M}_i$ with $\mathcal{M}'_i \subseteq \mathcal{M}_i$ for $i \in \{1, 2\}$. Consequently, it holds that $\mathcal{M}' := \mathcal{M}'_1 \cup \mathcal{M}'_2 \in \mathbb{M}_1 \otimes \mathbb{M}_2$ and $\mathcal{M}' \subseteq \mathcal{M}$. \square

Proofs about Computing Subsumer Justifications

Lemma 61 *Let \mathcal{T}_1 and \mathcal{T}_2 be normalised \mathcal{ELH} -terminologies, and let Σ be a signature. If $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$, then, for every $D \in \mathcal{EL}_\Sigma$ with $\mathcal{T}_1 \models X_1 \sqsubseteq D$, it holds that $\mathcal{T}_2 \models X_2 \sqsubseteq D$.*

Proof We prove by induction on the structure of D that for every $X_1, X_2 \in \mathbf{N}_C \cup \{\top\}$ with $\mathcal{T}_1 \models X_1 \sqsubseteq D$ and $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$, it holds that $\mathcal{T}_2 \models X_2 \sqsubseteq D$.

Let $D \in \mathcal{EL}_\Sigma$ and let $X_1, X_2 \in \mathbf{N}_C$ such that $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$ and $\mathcal{T}_1 \models X_1 \sqsubseteq D$. We now distinguish between the following cases.

- $D = \top$: then we are done, since $\mathcal{T}_2 \models X_2 \sqsubseteq \top$ always holds.

- $D \in \Sigma \cap \mathsf{N}_C$: then we have $\mathcal{T}_2 \models X_2 \sqsubseteq D$ by definition of a Σ -subsumer graph simulation.
- $D = \exists r.E$: then by Lemma 32 there exists an axiom $Y_1 \bowtie_1 \exists s.\psi_1 \in \mathcal{T}$ ($\bowtie_1 \in \{\sqsubseteq, \equiv\}$) such that $\mathcal{T} \models X_1 \sqsubseteq Y_1$, $\mathcal{T} \models \psi_1 \sqsubseteq E$ and $\mathcal{T} \models s \sqsubseteq r$. According to the Definition 60 there exists $Y_2 \bowtie_2 \exists s'.\psi_2 \in \mathcal{T}_2$ with $\mathcal{T}_2 \models X_2 \sqsubseteq Y_2$, $\bowtie_2 \in \{\sqsubseteq, \equiv\}$, $\mathcal{T} \models s' \sqsubseteq r$ and $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{E}\mathcal{L}}^{\vec{\cdot}} \langle \mathcal{T}_2, X_2 \rangle$. By applying the induction hypothesis we have $\mathcal{T} \models \psi_2 \sqsubseteq E$. Consequently, it holds that $\mathcal{T}_2 \models X_2 \sqsubseteq \exists r.E$.
- $D = D_1 \sqcap D_2$: then we have $\mathcal{T}_1 \models X_1 \sqsubseteq D_i$ for $i \in \{1, 2\}$ and by induction $\mathcal{T}_2 \models X_2 \sqsubseteq D_i$ for $i \in \{1, 2\}$, which finally yields $\mathcal{T}_2 \models X_2 \sqsubseteq D$.

□

Theorem 62 *Let \mathcal{T} be an acyclic, normalised $\mathcal{E}\mathcal{L}\mathcal{H}$ -terminology, and let Σ be a signature. Let $X \in \mathsf{N}_C$, and let $\mathbb{M}_X^{\vec{\cdot}} := \text{COVER}_{\rightarrow}(\mathcal{T}, X, \Sigma)$ computed by Algorithm 3. Then $\mathbb{M}_X^{\vec{\cdot}}$ is the set of all $\langle X, \Sigma \rangle$ -subsumer justifications of \mathcal{T} .*

Proof We first prove that $\mathbb{M}_X^{\vec{\cdot}}$ is a cover of the set of all $\langle X, \Sigma \rangle$ -subsumer module of \mathcal{T} by Definition 58. (i) We show that every $\mathcal{M} \in \mathbb{M}_X^{\vec{\cdot}}$ is an $\langle X, \Sigma \rangle$ -subsumer module of \mathcal{T} .

First, we define the relation $\succ_{\mathcal{T}} \subseteq \mathsf{N}_C \times \mathsf{N}_C$ as follows: for $X, Y \in \mathsf{N}_C$ we set $X \succ_{\mathcal{T}} Y$ iff there exists $X' \bowtie \exists s.Y \in \mathcal{T}$ with $\mathcal{T} \models X \sqsubseteq X'$, $\mathcal{T} \models s \sqsubseteq r$ and $\bowtie \in \{\sqsubseteq, \equiv\}$. Note that $\succ_{\mathcal{T}}$ is well-founded as \mathcal{T} is acyclic. The proof now proceeds by induction on $\succ_{\mathcal{T}}$.

Let $X \in \mathsf{N}_C$ and let $\mathcal{M} \in \mathbb{M}_X^{\vec{\cdot}}$. We now distinguish between the following cases.

- If there does not exist $Y \in \mathsf{N}_C$ with $X \succ_{\mathcal{T}} Y$, let $D \in \mathcal{E}\mathcal{L}_{\Sigma}$ such that $\mathcal{T} \models X \sqsubseteq D$. The proof now continues by induction on the structure of D . For $D = \top$ or $D = X$, it immediately follows that $\mathcal{M} \models X \sqsubseteq D$. For $D = B \in \mathsf{N}_C$ and $B \neq X$, there exists $\mathcal{J} \in \text{Just}_{\mathcal{T}}(X \sqsubseteq B)$ with $\mathcal{J} \subseteq \mathcal{M}$ (Line 4), i.e. $\mathcal{M} \models X \sqsubseteq D$ holds. For $D = D_1 \sqcap D_2$ we obtain $\mathcal{M} \models X \sqsubseteq D_1$ and $\mathcal{M} \models X \sqsubseteq D_2$ by applying the induction hypothesis on D_1 and D_2 . Consequently, it holds that $\mathcal{M} \models X \sqsubseteq D$. Note that D cannot be of the form $\exists r.E$ by Lemma 32.
- Otherwise, let $D \in \mathcal{E}\mathcal{L}_{\Sigma}$ such that $\mathcal{T} \models X \sqsubseteq D$. The proof now continues by induction on the structure of D . The cases of $D = \top$, $D = A \in \mathsf{N}_C$, and $D = D_1 \sqcap D_2$ can be proved as above. For $D = \exists r.E$. there exists $Y \bowtie \exists r.Z \in \mathcal{T}$ such that $\mathcal{T} \models X \sqsubseteq Y$ and $\mathcal{T} \models Z \sqsubseteq E$ by Lemma 32. Consequently, there exists $\alpha = Y' \bowtie_2 \exists S.Z' \in \mathcal{T}$ such that $\mathcal{T} \models X \sqsubseteq Y'$, $\mathcal{T} \models s \sqsubseteq r$, $\langle \mathcal{T}, Z \rangle \sim_{\Sigma, \mathcal{E}\mathcal{L}}^{\vec{\cdot}} \langle \mathcal{T}, Z' \rangle$ and $\alpha \in \mathcal{M}$. Note that $\langle \mathcal{T}, Z \rangle \sim_{\Sigma, \mathcal{E}\mathcal{L}}^{\vec{\cdot}} \langle \mathcal{T}, Z \rangle$ holds. Furthermore, there exists $\mathcal{J} \in \text{Just}_{\mathcal{T}}(X \sqsubseteq Y')$ such that $\mathcal{J} \subseteq \mathcal{M}$ and there exists $\mathcal{M}_{Z'} \in \text{COVER}_{\rightarrow}(\mathcal{T}, \Sigma, Z')$ such that $\mathcal{M}_{Z'} \subseteq \mathcal{M}$. As $X \succ Z'$, it follows from the induction hypothesis $\mathcal{M}_{Z'}$ is a $\langle Z', \Sigma \rangle$ -subsumer module of \mathcal{T} , i.e. $\mathcal{M}_{Z'} \models Z' \sqsubseteq E$. Additionally, it holds that $\mathcal{T} \models Z' \sqsubseteq E$ by $\langle \mathcal{T}, Z \rangle \sim_{\Sigma, \mathcal{E}\mathcal{L}}^{\vec{\cdot}} \langle \mathcal{T}, Z' \rangle$ and Lemma 61, which implies that $\mathcal{M}_{Z'} \models Z' \sqsubseteq E$. Overall, we hence obtain $\mathcal{M} \models X \sqsubseteq \exists r.E$.

(ii) We show that for every $\langle X, \Sigma \rangle$ -subsumer module \mathcal{M} of \mathcal{T} there exists $\mathcal{M}' \in \mathbb{M}_{\overline{X}}^{\rightarrow}$ such that $\mathcal{M}' \subseteq \mathcal{M}$.

The proof now proceeds by induction on \triangleright . Let $X \in \mathbb{N}_{\mathbb{C}}$ and let \mathcal{M} be a $\langle X, \Sigma \rangle$ -subsumer module of \mathcal{T} . We now distinguish between the following cases.

- If there does not exist $Y \in \mathbb{N}_{\mathbb{C}}$ with $X \succ_{\mathcal{T}} Y$, let $B \in \Sigma$. As \mathcal{M} is a $\langle X, \Sigma \rangle$ -subsumer module, there exists $S_B \in \text{Just}_{\mathcal{T}}(X \sqsubseteq B)$ with $S_B \subseteq \mathcal{M}$. Consequently, it is easy to see that there exists $\mathcal{M}' \in \mathbb{M}_{\overline{X}}^{\rightarrow}$ with $\mathcal{M}' = \bigcup_{B \in \Sigma} S_B$ as there does not exist $X' \bowtie \exists s.Y \in \mathcal{T}$ with $\mathcal{T} \models X \sqsubseteq X'$ and $\mathcal{T} \models s \sqsubseteq r$.
- Otherwise, let S be a Σ -subsumer simulation from \mathcal{T} to \mathcal{M} with $(X, X) \in S$. Similarly to the previous case, for every $B \in \Sigma$ there exists $S_B \in \text{Just}_{\mathcal{T}}(X \sqsubseteq B)$ with $S_B \subseteq \mathcal{M}$.

Now let $\{Y \bowtie \exists r.Z \in \mathcal{T} \mid r \in \Sigma, \mathcal{T} \models X \sqsubseteq Y \text{ and } \mathcal{T} \models s \sqsubseteq r\} = \{Y_1 \bowtie_{\exists r_1}.Z_1, \dots, Y_n \bowtie_{\exists r_n}.Z_n\}$ and let $1 \leq i \leq n$. Thus, by definition of S there exists $Y'_i \bowtie_{\exists s_i}.Z'_i \in \mathcal{M}$ such that $\mathcal{M} \models X \sqsubseteq Y'_i$ and $\mathcal{M} \models s_i \sqsubseteq r_i$ with $(Z, Z'_i) \in S$, i.e. $\langle \mathcal{T}, Z \rangle \sim_{\Sigma, \mathcal{ELH}}^{\rightarrow} \langle \mathcal{T}, Z'_i \rangle$ holds. Consequently, there exists $S_{Y'_i} \in \text{Just}_{\mathcal{T}}(X \sqsubseteq Y'_i)$ with $S_{Y'_i} \subseteq \mathcal{M}$ and $S_{r'_i} \in \text{Just}_{\mathcal{T}}(s_i \sqsubseteq r_i)$ with $S_{r'_i} \subseteq \mathcal{M}$. Moreover, as $(Z, Z'_i) \in S$, we have that \mathcal{M} is a $\langle Z'_i, \Sigma \rangle$ -subsumer module of \mathcal{T} . It follows from the induction hypothesis that there exists $\mathcal{M}'_{Z'_i} \in \mathbb{M}_{\overline{Z'_i}}^{\rightarrow} := \text{COVER}_{\rightarrow}(\mathcal{T}, \Sigma, Z'_i)$ with $\mathcal{M}'_{Z'_i} \subseteq \mathcal{M}$. Let

$$\mathcal{M}' := \bigcup_{B \in \Sigma} S_B \cup \bigcup_{i=1}^n (S_{Y'_i} \cup S_{r'_i} \cup \{Y'_i \bowtie_{\exists s_i}.Z'_i\} \cup \mathcal{M}'_{Z'_i}).$$

It is easy to see that $\mathcal{M}' \in \mathbb{M}_{\overline{X}}^{\rightarrow}$ and $\mathcal{M}' \subseteq \mathcal{M}$.

Then we have that $\mathbb{M}_{\overline{X}}^{\rightarrow}$ is the set of all $\langle X, \Sigma \rangle$ -subsumer justifications of \mathcal{T} . Let $\mathcal{M} \in \mathbb{M}_{\overline{X}}^{\rightarrow}$. We prove by assuming towards a contradiction. We assume that there exist an $\mathcal{M}' \subsetneq \mathcal{M}$ such that \mathcal{M}' is a $\langle X, \Sigma \rangle$ -subsumer module of \mathcal{T} . As $\mathbb{M}_{\overline{X}}^{\rightarrow}$ is initialed as $\{\emptyset\}$, we enumerate all possible situations that \mathcal{M} is generated in the algorithms but $\alpha \notin \mathcal{M}$, we have that the absence of α in any situation will lead to the failure of the subsumer simulation. So \mathcal{M}' could not be a $\langle X, \Sigma \rangle$ -subsumer module of \mathcal{T} , which contradicts to our assumption. □

Proofs about Computing Subsumee Justifications

Lemma 73 *Let \mathcal{T} be a normalised and acyclic \mathcal{ELH} -terminology, let Σ be a signature, and let $X \in \mathbb{N}_{\mathbb{C}}$ be pseudo-primitive in \mathcal{T} .*

Then X is not complex Σ -entailed w.r.t. \mathcal{T} .

Proof As X is pseudo-primitive in \mathcal{T} , we have $\text{non-conj}_{\mathcal{T}}(X) = \{X\}$. We can infer that there cannot exist $B \in \Sigma$ with $\mathcal{T} \models B \sqsubseteq Y$ and $\mathcal{T} \not\models B \sqsubseteq X$. Additionally, we observe that no axiom of the form $X \equiv \exists r.Y$ is contained in \mathcal{T} , which implies that X is not complex Σ -entailed w.r.t. \mathcal{T} . □

Lemma 74 *Let \mathcal{T} be a normalised and acyclic \mathcal{ELH} -terminology and let $X \in \mathbf{N}_C$ that is not complex Σ -entailed w.r.t. \mathcal{T} .*

Then for every $C \in \mathcal{EL}_\Sigma$ it holds that: $\mathcal{T} \models C \sqsubseteq X$ iff there exists $B \in \text{sub}_0(C)$ such that $\mathcal{T} \models B \sqsubseteq X$.

Proof Let $C \in \mathcal{EL}_\Sigma$.

First we prove that for every $C \in \mathcal{EL}_\Sigma$ it holds that: $\mathcal{T} \models C \sqsubseteq X$ only if there exists $B \in \text{sub}_0(C)$ such that $\mathcal{T} \models B \sqsubseteq X$. We assume that $\mathcal{T} \models C \sqsubseteq X$. Note that $\mathcal{T} \not\models \top \sqsubseteq C$ as \mathcal{T} is a normalised terminology, which implies that $\top \notin \text{sub}(\mathcal{T})$. Then, as X is not complex Σ -entailed w.r.t. \mathcal{T} , it follows that there exists $Y \in \text{non-conj}_{\mathcal{T}}(X)$ such that

- (i) for every $B \in \Sigma$ with $\mathcal{T} \models B \sqsubseteq Y$ it holds that $\mathcal{T} \models B \sqsubseteq X$, and
- (ii) for every $Y \equiv \exists r.Z \in \mathcal{T}$, r is Σ -entailed, it holds that Z is not Σ -entailed in \mathcal{T} .

Note that $\mathcal{T} \models C \sqsubseteq Y$ since $\mathcal{T} \models X \sqsubseteq Y$. We now distinguish between the following cases. If Y is pseudo-primitive in \mathcal{T} , then it follows from Lemma 31 that there exists $B \in \text{sub}_0(C)$ with $\mathcal{T} \models B \sqsubseteq Y$, which implies that $\mathcal{T} \models B \sqsubseteq X$ by (i).

Otherwise, there exists an axiom $Y \equiv \exists s.Y' \in \mathcal{T}$ and it holds that $\mathcal{T} \models C \sqsubseteq \exists s.Y'$. It follows from Lemma 31 that either (a) there exists $B \in \text{sub}_0(C)$ with $\mathcal{T} \models B \sqsubseteq Y$, or (b) there exists $\exists s.C' \in \text{sub}_0(C)$ with $\mathcal{T} \models C' \sqsubseteq Y'$. For (a), we have $\mathcal{T} \models B \sqsubseteq Y$, and thus, $\mathcal{T} \models B \sqsubseteq X$ by (i). We note that (b) cannot hold as s is Σ -entailed, $\mathcal{T} \models C' \sqsubseteq Y'$, and $\text{sig}(C') \subseteq \Sigma$, which implies that Y' would be Σ -entailed in \mathcal{T} .

“if” Then we prove that for every $C \in \mathcal{EL}_\Sigma$ it holds that: $\mathcal{T} \models C \sqsubseteq X$ if there exists $B \in \text{sub}_0(C)$ such that $\mathcal{T} \models B \sqsubseteq X$.

Let $B \in \text{sub}_0(C)$ such that $\mathcal{T} \models B \sqsubseteq X$. Since $\mathcal{T} \models C \sqsubseteq B$, it follows that $\mathcal{T} \models C \sqsubseteq X$. \square

Lemma 75 *Let \mathcal{T} be a normalised and acyclic \mathcal{ELH} -terminology and let $X \in \mathbf{N}_C$ be complex Σ -entailed w.r.t. \mathcal{T} .*

Then there exists $C \in \mathcal{EL}_\Sigma$ such that $\mathcal{T} \models C \sqsubseteq X$ and $\mathcal{T} \setminus \{\alpha\} \not\models C \sqsubseteq X$ for $\alpha = X \equiv D \in \mathcal{T}$.

Proof Let $\text{non-conj}_{\mathcal{T}}(X) = \{Y_1, \dots, Y_n\}$. As $X \in \mathbf{N}_C$ is complex Σ -entailed w.r.t. \mathcal{T} it holds that for every $1 \leq i \leq n$ one of the following conditions holds:

1. there exists $B_i \in \Sigma$ such that $\mathcal{T} \models B_i \sqsubseteq Y_i$ and $\mathcal{T} \not\models B_i \sqsubseteq X$, or
2. there exists $Y_i \equiv \exists r_i.Z_i \in \mathcal{T}$ such that r_i, Z_i is Σ -entailed in \mathcal{T} .

For every $1 \leq i \leq n$, we define

$$D_i := \begin{cases} B_i & \text{if } B_i \in \Sigma \text{ with } \mathcal{T} \models B_i \sqsubseteq Y_i, \text{ and } \mathcal{T} \not\models B_i \sqsubseteq X \\ \exists r_i.C_{Z_i} & \text{otherwise, where } C_{Z_i} \in \mathcal{EL}_\Sigma, \text{ with } \mathcal{T} \models C_{Z_i} \sqsubseteq Z_i \end{cases}$$

Finally, let $C := \prod_{i=1}^n D_i$. It is easy to see that $\mathcal{T} \models C \sqsubseteq X$. Note that there exists at most one axiom of the form $\alpha = X \equiv E \in \mathcal{T}$ as \mathcal{T} is a terminology, and that X is primitive in $\mathcal{T} \setminus \{\alpha\}$.

If we assume towards a contradiction that $\mathcal{T} \setminus \{\alpha\} \models C \sqsubseteq X$, it would follow from Lemma 31 that there exists $1 \leq j \leq n$ such that $\mathcal{T} \setminus \{\alpha\} \models B_j \sqsubseteq X$ and hence, $\mathcal{T} \models B_j \sqsubseteq X$; contradicting the assumptions. \square

Definition 76 Let \mathcal{T}_1 and \mathcal{T}_2 be normalised and acyclic \mathcal{ELH} -terminologies, and Σ be a signature. Additionally, let $X_1, X_2 \in \mathbb{N}_C$ such that $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\leftarrow} \langle \mathcal{T}_2, X_2 \rangle$. Then $\mathcal{M} \subseteq \mathcal{T}_2$ is a $\langle \mathcal{T}_1, X_1, \Sigma \rangle$ -subsumee module of \mathcal{T}_2 w.r.t. X_2 if $\mathcal{M} \models C \sqsubseteq X_2$ holds for every $C \in \mathcal{EL}_\Sigma$ such that $\mathcal{T}_1 \models C \sqsubseteq X_1$.

Lemma 67 Let \mathcal{T}_1 and \mathcal{T}_2 be two normalised \mathcal{ELH} -terminologies, and let Σ be a signature. If $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\leftarrow} \langle \mathcal{T}_2, X_2 \rangle$, then $\mathcal{T}_2 \models D \sqsubseteq X_2$ holds for every $D \in \mathcal{EL}_\Sigma$ with $\mathcal{T}_1 \models D \sqsubseteq X_1$.

Proof We prove that for every \mathcal{EL}_Σ -concept C and for every $(A_1, A_2) \in S$ with $\mathcal{T}_1 \models C \sqsubseteq A_1$ that $\mathcal{T}_2 \models C \sqsubseteq A_2$ holds by induction on the structure of C .

Let now C be an \mathcal{EL}_Σ -concept and let $\langle \mathcal{T}_1, A_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\leftarrow} \langle \mathcal{T}_2, A_2 \rangle$ such that $\mathcal{T}_1 \models C \sqsubseteq A_1$. We distinguish between the following cases:

- $C = \top$: this case cannot occur since $\mathcal{T}_1 \not\models C \sqsubseteq A_1$.
- $C = B \in \mathbb{N}_C$: as $B \in \Sigma$, we have $\mathcal{T}_2 \models C \sqsubseteq A_2$ by the definition 65.
- $C = \exists r.D$ for an \mathcal{EL}_Σ -concept D : we can infer that $\mathcal{T}_1 \models \exists r.D \sqsubseteq A_{1,i}$ for every $A_{1,i} \in \text{non-conj}_{\mathcal{T}_1}(A_1) = \{A_{1,1}, \dots, A_{1,n}\}$. Moreover, we can infer from Lemma 31 that every $A_{1,i} \in \text{non-conj}_{\mathcal{T}_1}(A_1)$ is not pseudo-primitive in \mathcal{T}_1 , i.e. for every $A_{1,i} \in \text{non-conj}_{\mathcal{T}_1}(A_1)$ we have $A_{1,i} \equiv \exists r.D_{1,i} \in \mathcal{T}_1$ with $\mathcal{T}_1 \models D \sqsubseteq D_{1,i}$ and r is Σ -entailed. In particular, for every $A_{1,i} \in \text{non-conj}_{\mathcal{T}_1}(A_1)$, $D_{1,i}$ is Σ -entailed.

Let now $\text{non-conj}_{\mathcal{T}_2}(A_2) = \{A_{2,1}, \dots, A_{2,m}\}$. It then follows from the definition 65 for every $1 \leq j \leq m$ that $A_{2,j} \equiv \exists r.D_{2,j} \in \mathcal{T}_2$ and that there exists $1 \leq i \leq n$ with $\langle \mathcal{T}_1, D_{1,i} \rangle \sim_{\Sigma, \mathcal{EL}}^{\leftarrow} \langle \mathcal{T}_2, D_{2,j} \rangle$, i.e. we obtain $\mathcal{T}_2 \models D \sqsubseteq D_{2,j}$ for every $1 \leq j \leq m$ by applying the induction hypothesis, and thus $\mathcal{T}_2 \models C \sqsubseteq A_2$.

- $C = C_1 \sqcap \dots \sqcap C_n$: we can infer that $\mathcal{T} \models C \sqsubseteq A'_1$ for every $A'_1 \in \text{non-conj}_{\mathcal{T}_1}(A_1)$, i.e. every $A'_1 \in \text{non-conj}_{\mathcal{T}_1}(A_1)$. Now, let $A'_2 \in \text{non-conj}_{\mathcal{T}_2}(A_2)$, and let $A'_1 \in \text{non-conj}_{\mathcal{T}_1}(A_1)$ with $\langle \mathcal{T}_1, A'_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\leftarrow} \langle \mathcal{T}_2, A'_2 \rangle$, which exists by the definition 65. It then follows from Lemma 31 that there exists $1 \leq j \leq n$ and a conjunct Y of C_j such that $Y \in \mathbb{N}_C$ or $Y = \exists r.D$ (for an \mathcal{ELH} -concept D), and $\mathcal{T}_1 \models Y \sqsubseteq A'_1$, i.e. $\mathcal{T}_1 \models C_j \sqsubseteq A'_1$. By applying the induction hypothesis we can infer that $\mathcal{T}_2 \models C_j \sqsubseteq A'_2$ holds, and thus $\mathcal{T}_2 \models C \sqsubseteq A'_2$. Hence, we have $\mathcal{T}_2 \models C \sqsubseteq A'_2$ for every $A'_2 \in \text{non-conj}_{\mathcal{T}_2}(A_2)$, i.e. $\mathcal{T}_2 \models C \sqsubseteq A_2$.

\square

Lemma 77 *Let \mathcal{T}_1 and \mathcal{T}_2 be normalised and acyclic \mathcal{ELH} -terminologies, and Σ be a signature. Additionally, let $X_1, X_2 \in \mathbf{N}_C$ such that $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\leftarrow} \langle \mathcal{T}_2, X_2 \rangle$, and let*

$$\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \text{COVER}_{\leftarrow}(\mathcal{T}_1, \mathcal{T}_2, \Sigma, X_1, X_2)$$

computed by Algorithm 4. Then for every $\mathcal{M} \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$, we have that for every \mathcal{EL}_Σ C , $\mathcal{T}_1 \models C \sqsubseteq X_1$ implies that $\mathcal{M} \models C \sqsubseteq X_2$.

Proof First, we define the relation $\triangleright_{\mathcal{T}_1} \subseteq \mathbf{N}_C \times \mathbf{N}_C$ as follows: for $X, Y \in \mathbf{N}_C$ we set $X \triangleright Y$ iff either there exists $X \equiv \exists r.Y \in \mathcal{T}_1$ or there exists $X \equiv Y_1 \sqcap \dots \sqcap Y_n \in \mathcal{T}_1$ such that $Y_i = Y$ for some $1 \leq i \leq n$. Note that $\triangleright_{\mathcal{T}_1}$ is well-founded as \mathcal{T}_1 is acyclic. The proof now proceeds by induction on $\triangleright_{\mathcal{T}_1}$.

Let $X_1, X_2 \in \mathbf{N}_C$ such that $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\leftarrow} \langle \mathcal{T}_2, X_2 \rangle$ and let $\mathcal{M} \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$. We now distinguish between the following cases.

If X_1 is not Σ -entailed w.r.t. \mathcal{T}_1 , then $\mathcal{M} = \emptyset$ and there does not exist $C \in \mathcal{EL}_\Sigma$ with $\mathcal{T}_1 \models C \sqsubseteq X_1$. Hence, it trivially holds for every $C \in \mathcal{EL}_\Sigma$ with $\mathcal{T}_1 \models C \sqsubseteq X_1$ that $\mathcal{M} \models C \sqsubseteq X_2$.

Otherwise, X_1 is Σ -entailed w.r.t. \mathcal{T}_1 . We further distinguish between the following cases. If X_1 is not complex Σ -entailed w.r.t. \mathcal{T}_1 , then let $C \in \mathcal{EL}_\Sigma$ such that $\mathcal{T}_1 \models C \sqsubseteq X_1$. Then by Lemma 74 there exists $B \in \text{sub}_0(C)$ such that $\mathcal{T}_1 \models B \sqsubseteq X_1$. As $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\leftarrow} \langle \mathcal{T}_2, X_2 \rangle$ there exists $J \in \text{Just}_{\mathcal{T}_2}(B \sqsubseteq X_2)$ with $J \subseteq \mathcal{M}$, which implies that $\mathcal{M} \models C \sqsubseteq X_2$.

Otherwise, X_1 is complex Σ -entailed w.r.t. \mathcal{T}_1 . We distinguish between the following cases. If $X_1 \equiv \exists r.Y_1 \in \mathcal{T}_1$ such that r or Y_1 is not Σ -entailed w.r.t. \mathcal{T}_1 , then let $C \in \mathcal{EL}_\Sigma$ such that $\mathcal{T}_1 \models C \sqsubseteq X_1$, and therefore $\mathcal{T}_1 \models C \sqsubseteq \exists r.Y_1$. By Lemma 31 we can infer that there exists $B \in \text{sub}_0(C)$ with $\mathcal{T}_1 \models B \sqsubseteq \exists r.Y_1$. We can then show that $\mathcal{M} \models C \sqsubseteq X_2$ as above.

If $X_1 \equiv \exists r.Y_1 \in \mathcal{T}_1$ such that $\mathcal{T}_1 \models s \sqsubseteq r, s \in \Sigma$ and Y_1 is Σ -entailed w.r.t. \mathcal{T}_1 , then let $C \in \mathcal{EL}_\Sigma$ such that $\mathcal{T}_1 \models C \sqsubseteq X_1$, and therefore $\mathcal{T}_1 \models C \sqsubseteq \exists r.Y_1$. By Lemma 31 either (a) there exists $B \in \text{sub}_0(C)$ with $\mathcal{T}_1 \models B \sqsubseteq \exists r.Y_1$, or (b) there exists $\exists r'.C' \in \text{sub}_0(C)$ with $\mathcal{T}_1 \models C' \sqsubseteq Y_1$ and $\mathcal{T}_1 \models r' \sqsubseteq s$. For (a), we can show that $\mathcal{M} \models C \sqsubseteq X_2$ as above.

For (b), we can infer that for every $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$ there exists $X'_2 \equiv \exists r'.Y'_2 \in \mathcal{T}_2$ such that $\mathcal{T}_2 \models r' \sqsubseteq s$ and $\langle \mathcal{T}_1, Y_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\leftarrow} \langle \mathcal{T}_2, Y'_2 \rangle$ as $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\leftarrow} \langle \mathcal{T}_2, X_2 \rangle$. Moreover, for every $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$ there exists $\mathcal{M}'_{Y'_2} \in \mathbb{M}_{(Y_1, Y'_2)}^{\leftarrow}$ (Line 8 in Algorithm 6) such that $\mathcal{M}'_{Y'_2} \cup \{X'_2 \equiv \exists r.Y'_2\} \subseteq \mathcal{M}$ (Line 10 in Algorithm 6).

By applying the induction hypothesis we have that for every \mathcal{EL}_Σ concept D , $\mathcal{T}_1 \models D \sqsubseteq Y'_2$ implies that $\mathcal{M}'_{Y'_2} \models D \sqsubseteq Y'_2$ for every $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$. Consequently, we have that $\mathcal{M} \models C' \sqsubseteq X'_2$ for every $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$. Finally, as $\text{max-tree}_{\mathcal{T}_2}^\square(X_2) \subseteq \mathcal{M}$ (Line 3 in Algorithm 6), we have $\mathcal{M} \models C \sqsubseteq X_2$.

If $X_1 \equiv Y_1 \sqcap \dots \sqcap Y_m \in \mathcal{T}_1$, then let $C \in \mathcal{EL}_\Sigma$ such that $\mathcal{T}_1 \models C \sqsubseteq X_1$, i.e. $\mathcal{T}_1 \models C \sqsubseteq X'_1$ for every $X'_1 \in \text{non-conj}_{\mathcal{T}_1}(X_1)$. As $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} \neq \emptyset$, we can infer that there exists $\Gamma \in \text{DefForest}_{\mathcal{T}_2}^\square(X_2)$ such that for every $X'_2 \in \text{leaves}(\Gamma)$ there exists $\xi(X'_2) \in \text{non-conj}_{\mathcal{T}_1}(X_1)$ with $\langle \mathcal{T}_1, X'_1 \rangle \sim_{\Sigma, \mathcal{EL}}^{\leftarrow} \langle \mathcal{T}_2 \setminus \delta_\Gamma, X'_2 \rangle$ and there exists $\mathcal{M}'_{X'_2} \in \text{COVER}_{\leftarrow}(\mathcal{T}_1, \mathcal{T}_2 \setminus \delta_\Gamma, \Sigma, \xi(X'_2), X'_2)$ with $\mathcal{M}'_{X'_2} \subseteq \mathcal{M}$ (Lines 4 to 17 in Algorithm 7).

By applying the induction hypothesis we have for every $X'_2 \in \text{leaves}(\Gamma)$ that $\mathcal{M}'_{X'_2}$ is a $\langle \mathcal{T}_1, \xi(X'_2), \Sigma \rangle$ -subsumee module of $\mathcal{T}_2 \setminus \delta_\Gamma$ w.r.t. X'_2 , i.e. $\mathcal{M}'_{X'_2} \models C \sqsubseteq X'_2$ and $\mathcal{M} \models C \sqsubseteq X'_2$ hold as $\mathcal{T}_1 \models C \sqsubseteq \xi(X'_2)$. Consequently, we have $\mathcal{M} \models C \sqsubseteq X_2$ as $\Gamma \subseteq \mathcal{M}$ (Line 16 in Algorithm 7). \square

Lemma 78 *Let \mathcal{T}_1 and \mathcal{T}_2 be normalised and acyclic $\mathcal{E}\mathcal{L}\mathcal{H}$ -terminologies, and Σ be a signature. Additionally, let $X_1, X_2 \in \mathbf{N}_C$ such that $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{E}\mathcal{L}}^{\leftarrow} \langle \mathcal{T}_2, X_2 \rangle$, and let*

$$\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \text{COVER}_{\leftarrow}(\mathcal{T}_1, \mathcal{T}_2, \Sigma, X_1, X_2)$$

computed by Algorithm 4. Moreover, let $\mathcal{M} \subseteq \mathcal{T}_2$. If $\mathcal{T}_1 \models C \sqsubseteq X_1$ implies $\mathcal{M} \models C \sqsubseteq X_2$ for every $\mathcal{E}\mathcal{L}_\Sigma$ concept C . Then we have that there exists $\mathcal{M}' \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ such that $\mathcal{M}' \subseteq \mathcal{M}$.

Proof First, we define the relation $\succ_{\mathcal{T}_1} \subseteq \mathbf{N}_C \times \mathbf{N}_C$ as follows: for $X, Y \in \mathbf{N}_C$ we set $X \succ Y$ iff either there exists $X \equiv \exists r.Y \in \mathcal{T}_1$ or there exists $X \equiv Y_1 \sqcap \dots \sqcap Y_n \in \mathcal{T}_1$ such that $Y_i = Y$ for some $1 \leq i \leq n$. Note that \succ is well-founded as \mathcal{T}_1 is acyclic.

The proof now proceeds by induction on \succ . Let $X_1, X_2 \in \mathbf{N}_C$ with $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{E}\mathcal{L}}^{\leftarrow} \langle \mathcal{T}_2, X_2 \rangle$ and let \mathcal{M} be a $\langle \mathcal{T}_1, X_1, \Sigma \rangle$ -subsumee module of \mathcal{T}_2 w.r.t. X_2 . Additionally, let S be a Σ -subsumee simulation from \mathcal{T}_1 to \mathcal{M} with $(X_1, X_2) \in S$.

First, we observe that for every $B \in \Sigma$ with $\mathcal{T}_1 \models B \sqsubseteq X_1$, there exists $J_{B \sqsubseteq X_1}^{\mathcal{M}} \in \text{Just}_{\mathcal{T}_2}(B \sqsubseteq X_2)$ with $J_{B \sqsubseteq X_1}^{\mathcal{M}} \subseteq \mathcal{M}$.

We now distinguish between the following cases. If X_1 is not Σ -entailed w.r.t. \mathcal{T}_1 , we set $\mathcal{M}' := \emptyset$. Then it holds that $\mathcal{M}' \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ (Line 3 in Algorithm 4) and $\mathcal{M}' \subseteq \mathcal{M}$.

Otherwise, X_1 is Σ -entailed w.r.t. \mathcal{T}_1 , and we proceed as follows. If X_1 is not complex Σ -entailed w.r.t. \mathcal{T}_1 , we set $\mathcal{M}' := \bigcup_{B \in \Sigma} J_{B \sqsubseteq X_1}^{\mathcal{M}}$. Then it holds that $\mathcal{M}' \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ (Line 5 in Algorithm 4 and Algorithm 5) and $\mathcal{M}' \subseteq \mathcal{M}$.

Otherwise, X_1 is complex Σ -entailed w.r.t. \mathcal{T}_1 , which implies that X_1 is not pseudo-primitive in \mathcal{T}_1 by Lemma 73. We now distinguish between the following two cases.

If $X_1 \equiv \exists r.Y_1 \in \mathcal{T}_1$, we analyse the following subcases.

If $r \notin \Sigma$ or Y_1 is not Σ -entailed w.r.t. \mathcal{T}_1 , then for $\mathcal{M}' := \bigcup_{B \in \Sigma} J_{B \sqsubseteq X_1}^{\mathcal{M}}$ it holds that $\mathcal{M}' \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ (Line 5 in Algorithm 4 and Algorithm 5) and $\mathcal{M}' \subseteq \mathcal{M}$.

Otherwise, we assume that Y_1 and r is Σ -entailed w.r.t. \mathcal{T}_1 , $\mathcal{T} \models s \sqsubseteq r$ and $s \in \Sigma$. By Condition $(S_{\exists}^{\rightarrow})$ we have that for every $X'_2 \in \text{non-conj}_{\mathcal{M}}(X_2)$ there exists $X'_2 \equiv \exists r.Y'_2 \in \mathcal{M}$ such that $(Y_1, Y'_2) \in S$.

Now we show that $\text{max-tree}_{\mathcal{T}_2}^{\square}(X_2) \subseteq \mathcal{M}$ by showing that $\text{max-tree}_{\mathcal{T}_2}^{\square}(X_2) = \text{max-tree}_{\mathcal{M}}^{\square}(X_2)$. It immediately holds that $\text{max-tree}_{\mathcal{M}}^{\square}(X_2) \subseteq \text{max-tree}_{\mathcal{T}_2}^{\square}(X_2)$ as $\mathcal{M} \subseteq \mathcal{T}_2$. Let $\alpha \in \text{max-tree}_{\mathcal{T}_2}^{\square}(X_2)$. If we assume towards a contradiction that $\alpha \notin \text{max-tree}_{\mathcal{M}}^{\square}(X_2)$, then there exists $\xi \in \text{leaves}(\text{max-tree}_{\mathcal{M}}^{\square}(X_2))$ such that $\xi \in \text{def}_{\mathcal{T}_2}^{\square}$ and $\xi \equiv \exists r.\psi \in \mathcal{M} \subseteq \mathcal{T}_2$. We have derived a contradiction.

Let $X'_2 \in \text{non-conj}_{\mathcal{M}}(X_2)$. By Condition $(S_{\exists}^{\rightarrow})$ there exists $\alpha_{X'_2} := X'_2 \equiv \exists r'. Y'_2 \in \mathcal{M}$ with $\mathcal{T}_2 \models s \sqsubseteq r'$, $s \in \Sigma$ and $\langle \mathcal{T}_1, Y_1 \rangle \sim_{\Sigma, \mathcal{E}, \mathcal{L}}^{\leftarrow} \langle \mathcal{T}_2, Y'_2 \rangle$. Consequently, it holds that for every $\mathcal{E}\mathcal{L}_{\Sigma}$ concept C , $\mathcal{T}_1 \models C \sqsubseteq Y_1$ implies $\mathcal{M} \models C \sqsubseteq Y'_2$. By applying the induction hypothesis there exists $\mathcal{M}'_{X'_2} \in \mathbb{M}_{(Y_1, Y'_2)}^{\leftarrow}$ such that $\mathcal{M}'_{X'_2} \subseteq \mathcal{M}$.

We set

$$\begin{aligned} \mathcal{M}' := & \bigcup_{\substack{B \in \Sigma \\ \mathcal{T}_2 \models B \sqsubseteq X_2}} J_{B \sqsubseteq X_2}^{\mathcal{M}} \cup \{\text{max-tree}_{\mathcal{T}_2}^{\square}(X_2)\} \cup \text{Just}_{\mathcal{T}_2}(s \sqsubseteq r') \\ & \cup \bigcup_{X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)} \mathcal{M}'_{X'_2} \cup \{\alpha_{X'_2}\}. \end{aligned}$$

Then it holds that $\mathcal{M}' \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ (Line 10 in Algorithm 4 and Algorithm 6) and $\mathcal{M}' \subseteq \mathcal{M}$.

If $X_1 \equiv Y_1 \sqcap \dots \sqcap Y_m \in \mathcal{T}_1$, then for every $X'_2 \in \text{non-conj}_{\mathcal{M}}(X_2)$ there exists $\xi(X'_2) \in \text{non-conj}_{\mathcal{T}_1}(X_1)$ such that $(\xi(X'_2), X'_2) \in \mathcal{S}$ by Condition $(S_{\sqcap}^{\leftarrow})$. Hence, for $\Gamma := \text{max-tree}_{\mathcal{M}}^{\square}(X_2)$ it holds that $\Gamma \in \text{DefForest}_{\mathcal{T}_2}^{\square}(X_2)$ and $\Gamma \subseteq \mathcal{M}$.

Moreover, for every $X'_2 \in \text{non-conj}_{\mathcal{M}}(X_2)$ it holds that for every $\mathcal{E}\mathcal{L}_{\Sigma}$ concept C , $\mathcal{T}_1 \models C \sqsubseteq \xi(X'_2)$ implies $\mathcal{M} \models C \sqsubseteq X'_2$.

By applying the induction hypothesis for every

$$X'_2 \in \text{non-conj}_{\mathcal{M}}(X_2),$$

there exists $\mathcal{M}'_{X'_2} \in \mathbb{M}_{(\xi(X'_2), X'_2)}^{\leftarrow}$ such that $\mathcal{M}'_{X'_2} \subseteq \mathcal{M}$. We set

$$\mathcal{M}' := \bigcup_{\substack{B \in \Sigma \\ \mathcal{T}_2 \models B \sqsubseteq X_1}} J_{B \sqsubseteq X_1}^{\mathcal{M}} \cup \Gamma \cup \bigcup_{X'_2 \in \text{non-conj}_{\mathcal{M}}(X_2)} \mathcal{M}'_{X'_2}.$$

Note that for every $X'_2 \in \text{non-conj}_{\mathcal{M}}(X_2)$, $\langle \mathcal{T}_1, \xi(X'_2) \rangle \sim_{\Sigma, \mathcal{E}, \mathcal{L}}^{\leftarrow} \langle \mathcal{T}_2 \setminus \delta_{\Gamma}, X'_2 \rangle$ holds as $(\xi(X'_2), X'_2) \in \mathcal{S}$ and $\mathcal{M} \subseteq \mathcal{T}_2 \setminus \delta_{\Gamma}$.

Then it holds that $\mathcal{M}' \in \mathbb{M}_{\Gamma}^{\leftarrow}$ and therefore, $\mathcal{M}' \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ (Algorithm 7 and Line 13 in Algorithm 4) and $\mathcal{M}' \subseteq \mathcal{M}$. □

Theorem 70 *Let \mathcal{T} be an acyclic, normalised $\mathcal{E}\mathcal{L}\mathcal{H}$ -terminology, $X \in \mathbb{N}_{\mathcal{C}}$ and Σ be a signature. Additionally, let $\mathbb{M}_X^{\leftarrow} := \text{COVER}_{\leftarrow}(\mathcal{T}, X, \Sigma, \mathcal{T}, X)$ computed by Algorithm 4. Then $\mathbb{M}_X^{\leftarrow}$ is the set of all $\langle X, \Sigma \rangle$ -subsumee justifications of \mathcal{T} .*

Proof By Lemma 77, Lemma 78 and Definition 58, we have that $\mathbb{M}_X^{\leftarrow}$ is a cover of the set of $\langle X, \Sigma \rangle$ -subsumee modules of \mathcal{T} . Let $\mathcal{M} \in \mathbb{M}_X^{\leftarrow}$.

Then we have that $\mathbb{M}_X^{\leftarrow}$ is a cover of the set of $\langle X, \Sigma \rangle$ -subsumer modules of \mathcal{T} . Let $\mathcal{M} \in \mathbb{M}_X^{\leftarrow}$. We prove by assuming towards a contradiction. We assume that there exist an $\mathcal{M}' \subsetneq \mathcal{M}$ such that \mathcal{M}' is a $\langle X, \Sigma \rangle$ -subsumee module of \mathcal{T} . As $\mathbb{M}_X^{\leftarrow}$ is initialed as $\{\emptyset\}$, we enumerate all possible situations

that \mathcal{M} is generated in the algorithms but $\alpha \notin \mathcal{M}$, we have that the absence of α in any situation will lead to the failure of the subsumee simulation. So \mathcal{M}' could not be a $\langle X, \Sigma \rangle$ -subsumee module of \mathcal{T} , which contradicts to our assumption. So $\mathbb{M}_X^{\leftarrow}$ is the set of all $\langle X, \Sigma \rangle$ -subsumee justifications. \square

3.2 Subsumption Justifications of \mathcal{ELH}^r -Terminologies

In this section, we extend the notion of subsumption justification to \mathcal{ELH}^r language. For a signature Σ , let $\Sigma^{\text{dom}} = \{ \text{dom}(t) \mid t \in \mathbb{N}_R \cap \Sigma \}$ and $\Sigma^{\text{ran}} = \{ \text{ran}(t) \mid t \in \mathbb{N}_R \cap \Sigma \}$ be the sets consisting of concepts of the form $\text{dom}(t)$ and $\text{ran}(t)$ for every role name t in Σ , respectively. Similar to Definition 53, we can define subsumption justifications for \mathcal{ELH}^r -terminology as follows.

Definition 79 (Subsumption Justification for \mathcal{ELH}^r -terminology) *Let Σ be a signature, \mathcal{T} a normalised \mathcal{ELH}^r -terminology, and let $r \in \mathbb{N}_R$, $s \in \mathbb{N}_R \cap \Sigma$ and $X \in \mathbb{N}_C \cup \{ \text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_R \}$. We say that*

- $\mathcal{M} \subseteq \mathcal{T}$ is an $\langle X, \Sigma \rangle$ -subsumee module of \mathcal{T} iff for every $D \in \mathcal{EL}_\Sigma^{\text{ran}}$ such that $\mathcal{T} \models D \sqsubseteq X$, $\mathcal{M} \models D \sqsubseteq X$.
- $\mathcal{M} \subseteq \mathcal{T}$ is an $\langle X, \Sigma \rangle$ -subsumer module \mathcal{M} of \mathcal{T} iff for every $C \in \mathcal{EL}_\Sigma$ such that $\mathcal{T} \models X \sqsubseteq C$, $\mathcal{M} \models X \sqsubseteq C$;
- $\mathcal{M} \subseteq \mathcal{T}$ is a $\langle r, \Sigma \rangle$ -role subsumption module of \mathcal{T} iff for every $s \in \Sigma$ such that $\mathcal{T} \models \alpha$ where α is of the form either $r \sqsubseteq s$ or $s \sqsubseteq r$, $\mathcal{M} \models \alpha$.

Additionally, a set \mathcal{M} is called an $\langle X, \Sigma \rangle$ -subsumption module of \mathcal{T} iff \mathcal{M} is an $\langle X, \Sigma \rangle$ -subsumee and $\langle X, \Sigma \rangle$ -subsumer module of \mathcal{T} . An $\langle X, \Sigma \rangle$ -subsumee (resp. subsumer, subsumption) justification is an $\langle X, \Sigma \rangle$ -subsumee (resp. subsumer, subsumption) module of \mathcal{T} that is minimal w.r.t. \subseteq . Similarly, a $\langle r, \Sigma \rangle$ -role subsumption justification of \mathcal{T} is an $\langle r, \Sigma \rangle$ -role subsumption module of \mathcal{T} that is minimal w.r.t. \subseteq .

We denote a $\langle X, \Sigma \rangle$ -subsumee (resp. subsumer, subsumption) justification as $\mathcal{J}_\mathcal{T}^{\leftarrow}(X, \Sigma)$ (resp. $\mathcal{J}_\mathcal{T}^{\rightarrow}(X, \Sigma)$, $\mathcal{J}_\mathcal{T}(X, \Sigma)$). Note that there may exist multiple $\langle X, \Sigma \rangle$ -(subsumer, subsumee) subsumption justifications. So we denote the set of all $\langle X, \Sigma \rangle$ -subsumee (resp. subsumer, subsumption) justifications as $\mathbb{J}_\mathcal{T}^{\leftarrow}(X, \Sigma)$ (resp. $\mathbb{J}_\mathcal{T}^{\rightarrow}(X, \Sigma)$, $\mathbb{J}_\mathcal{T}(X, \Sigma)$). Similarly, we denote a $\langle r, \Sigma \rangle$ -role subsumption justification of \mathcal{T} as $\mathcal{J}_\mathcal{T}(r, \Sigma)$ and the set of all $\langle r, \Sigma \rangle$ -role subsumption justification of \mathcal{T} as $\mathbb{J}_\mathcal{T}(r, \Sigma)$. The notations are the same as subsumption justification on \mathcal{ELH} -terminology.

According to the Definition 79, it is easy to compute subsumption justifications.

Theorem 80 *Let \mathcal{T} be a normalised \mathcal{ELH}^r -terminology, let Σ be a signature, and let $X \in \mathbb{N}_C$. Additionally, let $\mathbb{J}_\mathcal{T}^{\leftarrow}(X, \Sigma)$ (resp. $\mathbb{J}_\mathcal{T}^{\rightarrow}(X, \Sigma)$, $\mathbb{J}_\mathcal{T}(X, \Sigma)$) be the set of all $\langle X, \Sigma \rangle$ -subsumee justifications (resp. subsumer justification, subsumption justification). We have that*

$$\mathbb{J}_\mathcal{T}^{\rightarrow}(X, \Sigma) := \text{Minimise}_{\subseteq}(\mathbb{J}_\mathcal{T}^{\leftarrow}(X, \Sigma) \otimes \mathbb{J}_\mathcal{T}(X, \Sigma)).$$

Example 81 Let $\Sigma = \{X, Y, r\}$ and let $\mathcal{T} = \{\alpha_1, \alpha_2, \alpha_3\}$, where $\alpha_1 := \text{ran}(r) \sqsubseteq X$, $\alpha_2 := Y \sqsubseteq X$, $\alpha_3 := X \equiv \exists r.Z$. Then both sets, $\mathcal{M} = \{\alpha_1, \alpha_3\}$ and \mathcal{T} itself, are $\langle X, \Sigma \rangle$ -subsumee modules of \mathcal{T} . Moreover, \mathcal{M} is a $\langle X, \Sigma \rangle$ -subsumee justification of \mathcal{T} .

We obtain the following propositions from together with Definition 79. Recall that we use ‘ran’ as an abbreviation for the query language \mathcal{ELH}^r .

Proposition 82 Let \mathcal{T} be a normalised \mathcal{ELH}^r -terminology, Σ be a signature, $X \in \mathbb{N}_C \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}$ and $r \in \Sigma \cap \mathbb{N}_R$. Then the following statements hold:

- \mathcal{M} is an $\langle X, \Sigma \rangle$ -subsumer module of \mathcal{T} iff $X \notin \text{lhsWtn}_{\Sigma}^{\text{ran}}(\mathcal{T}, \mathcal{M})$;
- \mathcal{M} is an $\langle X, \Sigma \rangle$ -subsumee module of \mathcal{T} iff $X \notin \text{rhsWtn}_{\Sigma}^{\text{ran}}(\mathcal{T}, \mathcal{M})$;
- \mathcal{M} is an $\langle r, \Sigma \rangle$ -role subsumption module of \mathcal{T} iff $r \notin \text{roleWtn}_{\Sigma}(\mathcal{T}, \mathcal{M})$.

The algorithm for computing role subsumption justifications for \mathcal{ELH}^r -terminologies coincides with the algorithm for \mathcal{ELH} -terminologies, cf. Section 3.1.1. Now we present the algorithms for computing subsumer and subsumee justifications for \mathcal{ELH}^r -terminologies.

3.2.1 Computing Subsumer Justifications

Similar to the algorithm introduced in Section 3.1.2, we use a specific notion of a subsumer simulation to guide us when collecting the relevant axioms for subsumer justifications. First, we extend the definition of a subsumer simulation to \mathcal{ELH}^r -terminologies.

Definition 83 (Subsumer Simulation for \mathcal{ELH}^r -terminology) Let \mathcal{T}_1 and \mathcal{T}_2 be two normalised \mathcal{ELH}^r -terminologies, and let Σ be a signature. Moreover, let $N(\mathcal{T}, \Sigma) = \{X, \text{dom}(r), \text{ran}(r) \mid X, r \in (\text{sig}(\mathcal{T}) \cup \Sigma), X \in \mathbb{N}_C, r \in \mathbb{N}_R\}$, where \mathcal{T} ranges over terminologies. A relation $S \subseteq N(\mathcal{T}_1, \Sigma) \times N(\mathcal{T}_2, \Sigma)$ is a Σ -subsumer simulation from \mathcal{T}_1 to \mathcal{T}_2 iff the following conditions $(S_{\mathbb{N}_C}^{\rightarrow})$ and $(S_{\exists}^{\rightarrow})$ are both satisfied:

- $(S_{\mathbb{N}_C}^{\rightarrow})$ if $(X_1, X_2) \in S$, then for every $\varphi \in \Sigma \cup \Sigma^{\text{dom}}$ with $\mathcal{T}_1 \models X_1 \sqsubseteq \varphi$, it holds that $\mathcal{T}_2 \models X_2 \sqsubseteq \varphi$;
- $(S_{\exists}^{\rightarrow})$ if $(X_1, X_2) \in S$, $X'_1 \bowtie_1 \exists r.Y_1 \in \mathcal{T}_1$ with $\bowtie_1 \in \{\sqsubseteq, \equiv\}$ and $\mathcal{T}_1 \models X_1 \sqsubseteq X'_1$, then for every $s \in \Sigma$ with $\mathcal{T}_1 \models r \sqsubseteq s$, there exists $X'_2 \bowtie_2 \exists r'.Y_2 \in \mathcal{T}_2$ with $\bowtie_2 \in \{\sqsubseteq, \equiv\}$ such that $\mathcal{T}_2 \models X_2 \sqsubseteq X'_2$, $\mathcal{T}_2 \models r' \sqsubseteq s$ and $(Y_1, Y_2) \in S$.

We write $\mathcal{T}_1 \sim_{\Sigma, \text{ran}}^{\rightarrow} \mathcal{T}_2$ iff there exists a Σ -subsumer simulation S from \mathcal{T}_1 to \mathcal{T}_2 such that for every $\varphi \in (\Sigma \cap \mathbb{N}_C) \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}$: $(\varphi, \varphi) \in S$.

For $X_1, X_2 \in (\Sigma \cap \mathbb{N}_C) \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}$, we write $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \text{ran}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$ iff there exists a Σ -subsumer simulation S from \mathcal{T}_1 to \mathcal{T}_2 with $(X_1, X_2) \in S$ for which $\mathcal{T}_1 \sim_{\Sigma, \text{ran}}^{\rightarrow} \mathcal{T}_2$.

A subsumer simulation captures the set of subsumers in the following way: If a Σ -subsumer simulation from \mathcal{T}_1 to \mathcal{T}_2 contains the pair (X_1, X_2) , then X_2 entails w.r.t. \mathcal{T}_2 all subsumers of X_1 w.r.t. \mathcal{T}_1 that are formulated in the signature Σ . Formally, we obtain the following theorem. Note that the right-hand side of an \mathcal{ELH}^r -inclusion ranges over \mathcal{EL} -concepts (cf. Definition 4).

Lemma 84 *Let \mathcal{T}_1 and \mathcal{T}_2 be normalised \mathcal{ELH}^r -terminologies, Σ a signature and $X_1 \in \mathbf{N}_C \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}$. Additionally, let $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \text{ran}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$. Then, for every $D \in \mathcal{EL}_\Sigma$: $\mathcal{T}_1 \models X_1 \sqsubseteq D$ implies $\mathcal{T}_2 \models X_2 \sqsubseteq D$.*

We can update Algorithm 3 according to the new subsumer simulation and get an algorithm to compute subsumer justifications for \mathcal{ELH}^r -terminology, which is shown in Algorithm 8. Compared with Algorithm 3, additionally, we need to collect the justifications of the inclusion of the form $X \sqsubseteq \text{dom}(r)$ that follows from \mathcal{T} . Moreover, the condition of for-loop in Algorithm 3 changes according to the new subsumer simulation definition.

Algorithm 8: Computing the set of all Subsumer Justifications

```

1 function COVER $\rightarrow$  ( $\mathcal{T}, X, \Sigma$ )
2    $\mathbb{M}_X^{\rightarrow} = \{\emptyset\}$ 
3   for every  $\varphi \in (\Sigma \cap \mathbf{N}_C) \cup \{\text{dom}(r) \mid r \in \Sigma\}$  such that  $\mathcal{T} \models X \sqsubseteq \varphi$  do
4      $\mathbb{M}_X^{\rightarrow} := \mathbb{M}_X^{\rightarrow} \otimes \text{Just}_{\mathcal{T}}(X \sqsubseteq \varphi)$ 
5   end
6   for every  $Y \bowtie_1 \exists r.Z \in \mathcal{T}$  ( $\bowtie_1 \in \{\sqsubseteq, \equiv\}$ ) and  $s \in \Sigma \cap \mathbf{N}_R$  such that  $\mathcal{T} \models X \sqsubseteq Y$  and
    $\mathcal{T} \models r \sqsubseteq s$  do
7      $\mathbb{M}_{\exists s.Z}^{\rightarrow} := \{\emptyset\}$ 
8     for every  $s' \in \{s' \in \Sigma \cap \mathbf{N}_R \mid \mathcal{T} \models r \sqsubseteq s'\}$  do
9       for every  $Y' \bowtie_2 \exists r'.Z' \in \mathcal{T}$  ( $\bowtie_2 \in \{\sqsubseteq, \equiv\}$ ) such that  $\mathcal{T} \models X \sqsubseteq Y'$ ,  $\mathcal{T} \models r' \sqsubseteq s$ ,
       and  $\langle \mathcal{T}, Z \rangle \sim_{\Sigma, \text{ran}}^{\rightarrow} \langle \mathcal{T}, Z' \rangle$  do
10         $\mathbb{M}_{Z'}^{\rightarrow} := \text{COVER}_{\rightarrow}(\mathcal{T}, Z', \Sigma)$ 
11         $\mathbb{M}_{\exists s.Z}^{\rightarrow} := \mathbb{M}_{\exists s.Z}^{\rightarrow} \cup (\{\{Y' \bowtie_2 \exists r'.Z'\}\} \otimes \mathbb{M}_{Z'}^{\rightarrow} \otimes \text{Just}_{\mathcal{T}}(X \sqsubseteq Y') \otimes \text{Just}_{\mathcal{T}}(r' \sqsubseteq s))$ 
12      end
13    end
14     $\mathbb{M}_X^{\rightarrow} := \mathbb{M}_X^{\rightarrow} \otimes \mathbb{M}_{\exists s.Z}^{\rightarrow}$ 
15  end
16  return Minimise $\sqsubseteq$ ( $\mathbb{M}_X^{\rightarrow}$ )

```

Theorem 85 *Let \mathcal{T} be an acyclic, normalised \mathcal{ELH}^r -terminology, and let Σ be a signature. Additionally, let $X \in \mathbf{N}_C \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}$, and let $\mathbb{M}_X^{\rightarrow} := \text{COVER}_{\rightarrow}(\mathcal{T}, X, \Sigma)$ computed by Algorithm 8. Then $\mathbb{M}_X^{\rightarrow}$ is the set of $\langle X, \Sigma \rangle$ -subsumer justifications of \mathcal{T} .*

3.2.2 Computing Subsumee Justifications

Similar to the algorithm for computing subsumee justifications that presented in Section 3.1.3, we update the definition of Σ -entailment and Σ -subsumee simulation to deal with range restrictions.

The axioms of the form $\text{ran}(r) \sqsubseteq X$ might cause non-trivial entailments. For example, let $\mathcal{T}_1 = \{X \equiv \exists r.Y, Y \equiv A_1 \sqcap A_2\}$, $\mathcal{T}_2 = \mathcal{T}_1 \cup \{\text{ran}(r) \sqsubseteq A_1\}$, $\Sigma = \{X, A_1, A_2, r\}$. Then we have $\mathcal{T}_1 \models \exists r.(A_1 \sqcap A_2)$, however, as the axiom $\text{ran}(r) \sqsubseteq A_1 \in \mathcal{T}_2$, the conjunct A_1 of Y is already covered by α . Hence, when we define the notion of Σ -reachability, an additional parameter $\zeta \in \{\varepsilon\} \cup (\mathbb{N}_R \cap \Sigma)$ is used. We call this additional parameter the *context of a role*, i.e. an expression of the form $\text{ran}(\zeta)$. We treat ε as a special role name and set $\text{ran}(\varepsilon) = \top$. The set of all role contexts, in symbols \mathcal{C}^Σ , is defined as $\mathcal{C}^\Sigma = \{\varepsilon\} \cup (\mathbb{N}_R \cap \Sigma)$.

In the following, we extend the notions of Σ -entailment (Definition 63) and complex Σ -entailment (Definition 66) to $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies in order to deal with range restrictions.

Definition 86 (Σ -Entailment for $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -subsumption) *Let \mathcal{T} be a normalised $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminology and Σ be a signature. For $\zeta \in \mathbb{N}_R \cup \{\varepsilon\}$, $A \in \mathbb{N}_C$ and $s \in \mathbb{N}_R$, we say that:*

- (i) $A \in \mathbb{N}_C$ is $(\Sigma, \text{ran}, \zeta)$ -entailed in \mathcal{T} iff there is an $\mathcal{E}\mathcal{L}\Sigma$ -concept C such that $\mathcal{T} \models \text{ran}(\zeta) \sqcap C \sqsubseteq A$;
- (ii) $s \in \mathbb{N}_R$ is Σ -entailed in \mathcal{T} iff there exists $s' \in \mathbb{N}_R \cap \Sigma$ such that $\mathcal{T} \models s' \sqsubseteq s$.

We now define the notion of a *subsumee simulation* from \mathcal{T}_1 to \mathcal{T}_2 as a subset of $\text{sig}^{\mathbb{N}_C}(\mathcal{T}_1) \times \text{sig}^{\mathbb{N}_C}(\mathcal{T}_2) \times \mathcal{C}_{\mathcal{T}_1}^\Sigma$, where $\mathcal{C}_{\mathcal{T}_1}^\Sigma := \{\varepsilon\} \cup (\mathbb{N}_R \cap (\Sigma \cup \text{sig}(\mathcal{T}_1)))$ is the range of role contexts.

Definition 87 (Complex Σ -Entailment for $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminology) *Let Σ be a signature, \mathcal{T} a normalised and acyclic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminology, and $X \in \mathbb{N}_C \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}$.*

We say that X is complex $(\Sigma, \text{ran}, \zeta)$ -entailed w.r.t. \mathcal{T} iff for every $Y \in \text{non-conj}_{\mathcal{T}}(X)$ one of the following conditions hold:

- (i) *there exists $B \in \Sigma^{(\text{ran}, \zeta)}$ such that $\mathcal{T} \models B \sqsubseteq Y$ and $\mathcal{T} \not\models B \sqsubseteq X$;*
- (ii) *there exists $Y \equiv \exists r.Z \in \mathcal{T}$ and r and Z are each $(\Sigma, \text{ran}, \zeta)$ -entailed in \mathcal{T} .*

Otherwise, X is said to be simply $(\Sigma, \text{ran}, \zeta)$ -entailed.

The concept name X being complex Σ -entailed w.r.t. \mathcal{T} states that every conjunct of X is either defined as a Σ -entailed existential restriction (Condition (ii)) or it is entailed by a Σ -concept name, dom- or ran-concept by which X is not entailed.

Definition 88 (Subsumee Simulation for $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminology) *Let \mathcal{T}_1 and \mathcal{T}_2 be two normalised $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies, and let Σ be a signature. A relation $S \subseteq \text{sig}^{\mathbb{N}_C}(\mathcal{T}_1) \times \text{sig}^{\mathbb{N}_C}(\mathcal{T}_2) \times \mathcal{C}_{\mathcal{T}_1}^\Sigma$ is a Σ -subsumee simulation from \mathcal{T}_1 to \mathcal{T}_2 iff the following conditions hold:*

- $(S_{N_C}^{\leftarrow, \text{ran}})$ if $(X_1, X_2, \zeta) \in S$, then for each $\varphi \in \Sigma \cup \Sigma^{\text{dom}} \cup \{\text{ran}(\zeta) \mid \zeta \neq \varepsilon\}$ and for each $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$ with $\mathcal{T}_2 \models \text{ran}(\zeta) \sqsubseteq X'_2$, $\mathcal{T}_1 \models \varphi \sqsubseteq X_1$ implies $\mathcal{T}_2 \models \varphi \sqsubseteq X'_2$;
- $(S_{\exists}^{\leftarrow, \text{ran}})$ if $(X_1, X_2, \zeta) \in S$ and $X_1 \equiv \exists r. Y_1 \in \mathcal{T}_1$ such that $\mathcal{T}_1 \models s \sqsubseteq r$ for $s \in \Sigma$ and Y_1 is (Σ, s) -entailed in \mathcal{T}_1 , then for every $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$ not entailed by $\text{dom}(s)$ or $\text{ran}(\zeta)$ w.r.t. \mathcal{T}_2 , there exists $X'_2 \equiv \exists r'. Y_2 \in \mathcal{T}_2$ such that $\mathcal{T}_2 \models s \sqsubseteq r'$ and $(Y_1, Y_2, s) \in S$;
- $(S_{\Pi}^{\leftarrow, \text{ran}})$ if $(X_1, X_2, \zeta) \in S$ and $X_1 \equiv Y_1 \sqcap \dots \sqcap Y_n \in \mathcal{T}_1$, then for every $Y_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$ not entailed by $\text{ran}(\zeta)$ w.r.t. \mathcal{T}_2 , there exists $Y_1 \in \text{non-conj}_{\mathcal{T}_1}(X_1)$ not entailed by $\text{ran}(\zeta)$ w.r.t. \mathcal{T}_2 with $(Y_1, Y_2, \varepsilon) \in S$.

We write $\mathcal{T}_1 \sim_{\Sigma, \text{ran}}^{\leftarrow} \mathcal{T}_2$ iff there is a Σ -subsumee simulation S from \mathcal{T}_1 to \mathcal{T}_2 such that for every $A, r \in \Sigma$: $(A, A, \varepsilon) \in S$ and $(A, A, r) \in S$.

For $\zeta \in \Sigma \cap N_R$, we write $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2, X_2 \rangle$ iff there is a Σ -subsumee simulation S from \mathcal{T}_1 to \mathcal{T}_2 with $(X_1, X_2, \zeta) \in S$ for which $\mathcal{T}_1 \sim_{\Sigma, \text{ran}}^{\leftarrow} \mathcal{T}_2$.

Lemma 89 Let \mathcal{T}_1 and \mathcal{T}_2 be normalised and acyclic $\mathcal{EL}\mathcal{H}^r$ -terminologies, and let Σ be a signature. Then: $\mathcal{T}_1 \sim_{\Sigma, \text{ran}}^{\leftarrow} \mathcal{T}_2$ iff $\text{rhsWtn}_{\Sigma}^{\text{ran}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$.

Lemma 90 Let \mathcal{T}_1 and \mathcal{T}_2 be two normalised and acyclic $\mathcal{EL}\mathcal{H}^r$ -terminologies, and let Σ be a signature. It holds that $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2, X_2 \rangle$ iff for every $C \in \mathcal{EL}_{\Sigma}$: $\mathcal{T}_1 \models \text{ran}(\zeta) \sqcap C \sqsubseteq X_1$ implies $\mathcal{T}_2 \models \text{ran}(\zeta) \sqcap C \sqsubseteq X_2$.

Next we devise the following algorithms for computing subsumee justifications for $\mathcal{EL}\mathcal{H}^r$ -terminologies. The notions that we use in the algorithm are the same as the same notions we introduced in Section 3.1.3 unless we revised them in this section.

Based on the notion of a subsumee simulation, we present Algorithm 9 for Computing the set of the subsumee justifications for a given $\mathcal{EL}\mathcal{H}^r$ -terminology \mathcal{T} , a concept $X \in N_C$ and a signature Σ . The function call for obtaining the $\langle X, \Sigma \rangle$ -subsumee justifications of \mathcal{T} is $\text{COVER}_{\leftarrow}(\mathcal{T}, X, \Sigma, \mathcal{T}, X, \varepsilon)$.

The following theorem states that Algorithm 9 can be used to compute a cover of the set of subsumee modules, which outputs all subsumee justifications via the operator $\text{Minimise}_{\subseteq}(\cdot)$.

Theorem 91 Let \mathcal{T} be an acyclic, normalised $\mathcal{EL}\mathcal{H}^r$ -terminology, $X \in N_C \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}$ and Σ be a signature. Additionally, let

$$\mathbb{M} := \text{Minimise}_{\subseteq}(\text{COVER}_{\leftarrow}(\mathcal{T}, X, \Sigma, \mathcal{T}, X, \varepsilon) \otimes \bigotimes_{r \in \Sigma \cap N_R} \text{COVER}_{\leftarrow}(\mathcal{T}, X, \Sigma, \mathcal{T}, X, r))$$

computed recursively by Algorithm 9. Then \mathbb{M} is the set of all $\langle X, \Sigma \rangle$ -subsumee justifications of \mathcal{T} .

Algorithm 9: Computing the set of all Subsumee Justifications for an \mathcal{ELH}^r -terminology

```

1 function COVER $_{\leftarrow}$ ( $\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \zeta$ )
2   if  $X_1$  is not  $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed w.r.t.  $\mathcal{T}_1$  then
3     | return  $\{\emptyset\}$ 
4   end
5    $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \text{COVER}_{\leftarrow}^{\text{Nc}}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \zeta)$ 
6   if  $X_1$  is not complex  $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed in  $\mathcal{T}_1$  then
7     | return  $\text{Minimise}_{\subseteq}(\mathbb{M}_{(X_1, X_2)}^{\leftarrow})$ 
8   end
9   if  $X_1 \equiv \exists r.Y \in \mathcal{T}_1$ , and  $r$  is  $\Sigma$ -entailed and  $Y$  is  $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed w.r.t.  $\mathcal{T}_1$  then
10    |  $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \mathbb{M}_{(X_1, X_2)}^{\leftarrow} \otimes \text{COVER}_{\leftarrow}^{\exists}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \zeta)$ 
11   end
12   else if  $X_1 \equiv Y_1 \sqcap \dots \sqcap Y_m \in \mathcal{T}_1$  then
13    |  $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \mathbb{M}_{(X_1, X_2)}^{\leftarrow} \otimes \text{COVER}_{\leftarrow}^{\sqcap}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \zeta)$ 
14   end
15   return  $\text{Minimise}_{\subseteq}(\mathbb{M}_{(X_1, X_2)}^{\leftarrow})$ 

```

Similar to \mathcal{ELH}^r -terminology \mathcal{T} , our algorithm for computing subsumption justifications in \mathcal{ELH}^r -terminology runs in time exponential in the size of \mathcal{T} and signature.

The idea of the proof of the lemmas and theorems in this section is similar with Section 3.1.4. The difference is that we use Lemma 34 instead of Lemma 31 to take care of role restrictions in \mathcal{ELH}^r -terminology.

Proposition 92 *Let \mathcal{T} be a normalised \mathcal{ELH}^r -terminology, Σ a signature, $X \in \text{Nc}$, and $\varphi \in \text{Nc} \cup \{\text{dom}(r), \text{ran}(r) \mid r \in \text{Nr}\}$. The algorithm for computing all $\langle \varphi, \Sigma \rangle$ -subsumer justifications and all $\langle X, \Sigma \rangle$ -subsumer justifications runs in EXPTIME in the size of \mathcal{T} .*

Algorithm 10: Computing the set of all Subsumee Projection Justifications for an \mathcal{ELH}^r -terminology ($\mathcal{S}_{\text{Nc}}^{\leftarrow, \mathcal{L}}$)

```

1 function COVER $_{\leftarrow}^{\text{Nc}}$  ( $\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \zeta$ )
2    $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} = \{\emptyset\}$ 
3   for every  $B \in \Sigma \cup \Sigma^{\text{dom}} \cup \{\text{ran}(\zeta) \mid \zeta \neq \varepsilon\}$  such that  $\mathcal{T}_1 \models B \sqsubseteq X_1$  do
4     | for every  $X_2 \in \text{non-conj}_{\mathcal{T}_2}(X_1)$  such that  $\zeta = \varepsilon$  or  $\mathcal{T}_2 \models \text{ran}(\zeta) \sqsubseteq X_2$  do
5       | |  $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \mathbb{M}_{(X_1, X_2)}^{\leftarrow} \otimes \text{Just}_{\mathcal{T}_2}(B \sqsubseteq X_2)$ 
6     | end
7   end
8   return  $\text{Minimise}_{\subseteq}(\mathbb{M}_{(X_1, X_2)}^{\leftarrow})$ 

```

Algorithm 11: Computing the set of all Subsumee Projection Justifications for an \mathcal{ELH}^r -terminology $(S_{\exists}^{\leftarrow, \mathcal{L}})$

```

1 function COVER $_{\leftarrow}^{\exists}$  ( $\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \zeta$ )
2   Let  $\alpha_{X_1} := X_1 \equiv \exists r.Y_1 \in \mathcal{T}_1$ 
3    $M_{(X_1, X_2)}^{\leftarrow} := \{\text{max-tree}_{\mathcal{T}_2}^{\square}(X_2)\}$ 
4   for every  $s \in \Sigma \cap N_R$  such that  $\mathcal{T}_1 \models s \sqsubseteq r$  do
5     for every  $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$  such that  $\zeta \neq \varepsilon$  implies  $\mathcal{T}_2 \not\models \text{ran}(\zeta) \sqsubseteq X'_2$  and
6        $\mathcal{T}_2 \not\models \text{dom}(s) \sqsubseteq X'_2$  do
7         Let  $\alpha_{X'_2} := X'_2 \equiv \exists r'.Y'_2 \in \mathcal{T}_2$   $M_{Y'_2}^{\leftarrow} := \text{COVER}_{\leftarrow}(\mathcal{T}_1, Y_1, \Sigma, \mathcal{T}_2, Y'_2, s)$ 
8          $M_{(X_1, X_2)}^{\leftarrow} := M_{(X_1, X_2)}^{\leftarrow} \otimes \{\{\alpha_{X'_2}\}\} \otimes \text{Just}_{\mathcal{T}_2}(s \sqsubseteq r) \otimes M_{Y'_2}^{\leftarrow}$ 
9       end
10    end
11  return Minimise $_{\subseteq}(M_{(X_1, X_2)}^{\leftarrow})$ 

```

3.3 Evaluation

We have implemented our algorithms for computing subsumption justifications and for computing all minimal modules for \mathcal{ELH} -terminology in Java. In the evaluation, we first show the feasibility of computing subsumption justifications; The tool we use to compute justifications is BEACON [2].

We have evaluated the performance of the implementation using two prominent biomedical ontologies: Snomed CT (version Jan 2016), an \mathcal{ELH} -terminology consisting of 317891 axioms, and NCI (version 16.03d), an \mathcal{EL} -terminology containing 165341 axioms. All the experiments were conducted on the machines equipped with Intel Xeon Core 4 Duo CPU running at 2.50GHz and with 64GiB RAM. An execution timeout is 10 minutes.

Moreover, Table 3.1 details how the computation time was spent on different sub-tasks which determined the bottleneck of our tool. Indeed, 94.6% of the computation time was spent by BEACON on computing all justifications for concept name inclusions. Therefore, a considerable boost in performance of our tool can be expected by precomputing such justifications.

Computation of all Subsumption Justifications Table 3.2 shows the results obtained for computing all subsumption justifications. The first row indicates the ontology used in each experiment. The experiments are divided into four categories according to the numbers of concept and role names included in an input signature, as specified in the second row. For each category, we generated 1000 random signatures and computed the corresponding subsumption justifications for each concept name in the signature. Row 3 shows that multiple subsumption justifications can exist in real-world ontologies, e.g., there are 1328 subsumption justifications for a random signature consisting of 30 concept and 10 role names in Snomed CT. Meanwhile, Row 4 reports the cardinality of subsumption justifications, e.g., the largest one having 27 axioms for a signature of 30 concept and 10 role names from NCI. Row 5 shows that the subsumption justifications for more than 82.4% of random signatures

Algorithm 12: Computing the set of all Subsumee Justifications for an \mathcal{ELH}^r -terminology $(S_{\sqcap}^{\leftarrow, \mathcal{L}})$

```

1 function COVER $_{\leftarrow}^{\sqcap}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \zeta)$ 
2   let  $\alpha_{X_1} := X_1 \equiv Y_1 \sqcap \dots \sqcap Y_m \in \mathcal{T}_1$ 
3    $M_{(X_1, X_2)}^{\leftarrow} := \emptyset$ 
4   for every  $\Gamma \in \text{DefForest}_{\mathcal{T}_2}^{\sqcap}(X_2)$  do
5     Let  $\delta_{\Gamma} := \{ \text{def}_{\mathcal{T}_2}^{\sqcap}(X') \mid X' \in \text{leaves}(\Gamma) \cap \text{def}_{\mathcal{T}_2}^{\sqcap} \}$ 
6      $M_{\Gamma}^{\leftarrow} := \{ \Gamma \}$ 
7     for every  $X'_2 \in \text{leaves}(\Gamma)$  such that  $\zeta = \varepsilon$  or  $\mathcal{T}_2 \not\models \text{ran}(\zeta) \sqsubseteq X'_2$  do
8        $M_{X'_2}^{\leftarrow} := \emptyset$ 
9       for every  $X'_1 \in \text{non-conj}_{\mathcal{T}_1}(X_1)$  such that  $\zeta = \varepsilon$  or  $\mathcal{T}_2 \not\models \text{ran}(\zeta) \sqsubseteq X'_1$  do
10        if  $\langle \mathcal{T}_1, X'_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow} \langle \mathcal{T}_2 \setminus \delta_{\Gamma}, X'_2 \rangle$  then
11           $M_{X'_2}^{\leftarrow} := M_{X'_2}^{\leftarrow} \cup \text{COVER}_{\leftarrow}(\mathcal{T}_1, X'_1, \Sigma, \mathcal{T}_2 \setminus \delta_{\Gamma}, X'_2, \varepsilon)$ 
12        end
13      end
14       $M_{\Gamma}^{\leftarrow} := M_{\Gamma}^{\leftarrow} \otimes M_{X'_2}^{\leftarrow}$ 
15    end
16     $M_{(X_1, X_2)}^{\leftarrow} := M_{(X_1, X_2)}^{\leftarrow} \cup M_{\Gamma}^{\leftarrow}$ 
17  end
18  return  $\text{Minimise}_{\subseteq}(M_{(X_1, X_2)}^{\leftarrow})$ 

```

Sub-Task	JUST	Reasoner	Simulation Check	others
Percentage (%)	94.60	1.79	1.57	2.04

Table 3.1 Percentage of computation time consumed by sub-task of the algorithm for computing subsumption justifications

can be computed within 10 mins, whereas the statistics of the actual computation times is given in Row 6. More detailed results are shown in Table 3.3.

Ontologies $(\Sigma \cap N_C , \Sigma \cap N_R)$	Snomed CT		NCI	
	(10,10)	(30,10)	(10,10)	(30,10)
Nb. of all Subsumption Justifications	1.0 / 19.0 / 1.0 / 0.9	1.0 / 1328.0 / 1.0 / 10.7	1.0 / 136.0 / 1.0 / 3.5	1.0 / 7008.0 / 1.0 / 41.8
Card. of a Subsumption Justification	0.0 / 18.0 / 0.0 / 1.7	0.0 / 15.0 / 0.0 / 3.2	0.0 / 15.0 / 0.0 / 3.2	0.0 / 27.0 / 0.0 / 8.1
Success Rate	88.7%	82.4%	84.8%	91.7%
Computation Time (s)	0.2 / 519.7 / 0.4 / 59.2	0.7 / 576.3 / 1.6 / 28.5	0.2 / 472.4 / 1.3 / 66.9	0.2 / 577.3 / 7.6 / 97.0

Table 3.2 The statistics of experiments on computing all subsumption justifications for signatures generated at random, 1000 signatures of each size (minimal / maximal / median / standard deviation)

Ontologies	Snomed CT			NCI		
	$ \Sigma \cap N_c $	10	30	10	10	30
	$ \Sigma \cap N_r $	10	10	10	10	10
	Success Rate (%)	88.7	82.4	84.8	91.7	91.7
Number	Subsumer Justifications	1.0 / 19.0 / 1.0 / 1.0	1.0 / 1596.0 / 1.0 / 10.2	1.0 / 136.0 / 1.0 / 3.8	1.0 / 1.0 / 162.0 / 1.0 / 3.7	1.0 / 7008.0 / 1.0 / 41.7
	Subsumee Justifications	1.0 / 7.0 / 1.0 / 1.0	1.0 / 1440.0 / 1.0 / 10.4	1.0 / 9.0 / 1.0 / 0.1	1.0 / 9.0 / 1.0 / 0.1	1.0 / 7008.0 / 1.0 / 41.7
	Subsumption Justifications	1.0 / 19.0 / 1.0 / 0.9	1.0 / 1328.0 / 1.0 / 10.7	1.0 / 136.0 / 1.0 / 3.5	1.0 / 7008.0 / 1.0 / 41.8	1.0 / 7008.0 / 1.0 / 41.8
Ratio of number of subsumption covers that contain more than 1 justification to the total number of subsumption covers computed within the timeout (%)						
		20.4	21.5	40.4	47.0	47.0
Size	Subsumer Justifications	0.0 / 26.0 / 0.0 / 2.2	0.0 / 15.0 / 0.0 / 3.3	1.0 / 19.0 / 1.0 / 1.0	0.0 / 16.0 / 0.0 / 3.3	0.0 / 16.0 / 0.0 / 3.3
	Subsumee Justifications	0.0 / 26.0 / 0.0 / 0.6	0.0 / 44.0 / 0.0 / 10.1	0.0 / 8.0 / 0.0 / 0.2	0.0 / 27.0 / 0.0 / 9.4	0.0 / 27.0 / 0.0 / 9.4
	Subsumption Justifications	0.0 / 18.0 / 0.0 / 1.7	0.0 / 15.0 / 0.0 / 3.2	0.0 / 15.0 / 0.0 / 3.2	0.0 / 27.0 / 0.0 / 8.1	0.0 / 27.0 / 0.0 / 8.1
Rate of Signatures that have non-empty Subsumption Justifications(%)						
		19.9	30.7	70.8	92.5	92.5
Rate of Concept names that have non-empty Subsumption Justifications(%)						
		20.4	21.5	40.3	47.0	47.0
Time (s)	Subsumer Justifications	0.0 / 490.6 / 0.0 / 18.6	0.0 / 499.0 / 0.0 / 14.0	0.0 / 471.1 / 0.0 / 22.1	0.0 / 574.5 / 0.0 / 18.9	0.0 / 574.5 / 0.0 / 18.9
	Subsumee Justifications	0.0 / 0.3 / 0.0 / 0.0	0.0 / 23.8 / 0.0 / 0.2	0.0 / 17.2 / 0.0 / 0.2	0.0 / 37.6 / 0.0 / 0.3	0.0 / 37.6 / 0.0 / 0.3
	Subsumption Justifications	0.2 / 519.7 / 0.4 / 59.2	0.7 / 576.3 / 1.6 / 28.5	0.2 / 472.4 / 1.3 / 66.9	0.2 / 577.3 / 7.6 / 97.0	0.2 / 577.3 / 7.6 / 97.0

Table 3.3 Computation of numbers, size and computation time of subsumer/subsumee justifications for Snomed CT and NCI for 1000 random signatures containing 10/30 concept names and 10 role names (minimal / maximal / median / standard deviation) with timeout of 10 mins

Chapter 4

Extraction of Ontology Subsumption Modules

In this section, we first introduce a notion of ontology module called *basic subsumption module w.r.t. a given signature* that preserves all \mathcal{ELH}^r -inclusions of a terminology over the signature. Note that this is a different notion from $\langle X, \Sigma \rangle$ -subsumption modules that are w.r.t. a particular concept name X and a signature.

We present two approaches to computing all minimal subsumption modules of an ontology for a signature: a black-box approach and a glass-box approach. We follow the black-box approach for finding one or all justifications by replacing the entailment tests with logical difference checks, obtaining modules that preserve not only a given consequence but all entailments over a signature. The glass-box algorithm computes all possible subsumption modules of an ontology w.r.t. the signature by combining all subsumption justifications of every concept name in signature, as discussed in the previous chapter. At the end, we evaluate both approaches by computing all subsumption modules on large biomedical ontologies. The experiment shows that either black-box algorithm or glass-box algorithm is practical to be used to compute basic subsumption modules in large ontologies when the signature is not relatively large. In general, glass-box algorithm is faster than black-box algorithm. However, glass-box algorithm can only deal with acyclic terminologies for the moment.

4.1 Definition of Subsumption Module

In general, a module is a subset of an ontology that can act as a substitute for the ontology w.r.t. a given signature. In this chapter, we consider the notion of basic modules from [10] for \mathcal{ELH}^r -terminologies. We now give a formal definition of the module notion.

Definition 93 (Basic \mathcal{ELH}^r -Subsumption Module) *Let \mathcal{T} be an \mathcal{ELH}^r -terminology and let Σ be a signature. A subset $\mathcal{M} \subseteq \mathcal{T}$ is called a basic \mathcal{ELH}^r -subsumption module of \mathcal{T} w.r.t. Σ iff for*

all $\mathcal{EL}\mathcal{H}^r$ -inclusions α with $\text{sig}(\alpha) \subseteq \Sigma$ it holds that $\mathcal{T} \models \alpha$ iff $\mathcal{M} \models \alpha$. \mathcal{M} is called a minimal basic $\mathcal{EL}\mathcal{H}^r$ -subsumption module of \mathcal{T} w.r.t. Σ iff \mathcal{M} is minimal w.r.t. \subseteq with that property.

In the remainder, for simplicity's sake, we refer to a (minimal) basic $\mathcal{EL}\mathcal{H}^r$ -subsumption module of \mathcal{T} w.r.t. Σ as a (minimal) basic subsumption module.

Lemma 94 *Let \mathcal{T} be an $\mathcal{EL}\mathcal{H}^r$ -terminology, and let Σ be a signature. Then $\mathcal{M} \subseteq \mathcal{T}$ is a basic $\mathcal{EL}\mathcal{H}^r$ -subsumption module of \mathcal{T} w.r.t. Σ iff $\text{cDiff}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{M}) = \emptyset$, where $\mathcal{L} = \mathcal{EL}\mathcal{H}^r$.*

Every subset \mathcal{M} of a terminology \mathcal{T} that preserves the entailment of all $\mathcal{EL}\mathcal{H}^r$ -subsumptions over a given signature Σ is a basic $\mathcal{EL}\mathcal{H}^r$ -subsumption module of \mathcal{T} w.r.t. Σ . In particular, \mathcal{T} itself is a basic $\mathcal{EL}\mathcal{H}^r$ -subsumption module of \mathcal{T} w.r.t. any signature. It can readily be seen that \mathcal{M} is a basic $\mathcal{EL}\mathcal{H}^r$ -subsumption module of \mathcal{T} w.r.t. Σ iff $\text{cDiff}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{M}) = \emptyset$, for the query language $\mathcal{L} = \mathcal{EL}\mathcal{H}^r$ (cf. Definition 19). We have that \mathcal{M} and \mathcal{T} are inseparable w.r.t. $\mathcal{EL}\mathcal{H}^r$ -inclusions over Σ . More precisely, as $\mathcal{M} \subseteq \mathcal{T}$, it holds that \mathcal{T} is a conservative extension of \mathcal{M} w.r.t. $\mathcal{EL}\mathcal{H}^r$ -inclusions over Σ [35].

The following propositions show the relation between the basic $\mathcal{EL}\mathcal{H}^r$ -subsumption module notion and other module notions.

A relation between basic $\mathcal{EL}\mathcal{H}^r$ -subsumption modules and depleting modules is given below:

Proposition 95 *Let \mathcal{T} be an $\mathcal{EL}\mathcal{H}^r$ -TBox and Σ a signature. Let \mathcal{M}^{\dagger} be a depleting module of \mathcal{T} w.r.t. Σ . Then \mathcal{M}^{\dagger} is a basic subsumption module of \mathcal{T} w.r.t. Σ .*

Proof By Property 41, we have that a depleting Σ -module \mathcal{M}^{\dagger} of \mathcal{T} , is also a plain Σ -module of \mathcal{T} . As model-theoretic inseparability is strictly stronger than inseparability w.r.t. first-order logic queries (see, e.g., [26] and references therein), \mathcal{M}^{\dagger} is a basic subsumption module w.r.t. Σ . \square

There may be exponentially many (in the size of \mathcal{T}) basic subsumption modules (see Example 99). For the use-case of ontology re-use, however, we are most interested in modules that are as small as possible [16]. Note that smallest modules (regarding the number of axioms) are also minimal w.r.t. \subseteq , whereas the converse does not hold in general, i.e., there may be minimal modules w.r.t. \subseteq that contain more axioms than other minimal modules w.r.t. \subseteq .

Example 96 *Let $\mathcal{T} = \{A \sqsubseteq X \sqcap Y, X \sqsubseteq B, Y \sqsubseteq Z, Z \sqsubseteq B\}$ be an $\mathcal{EL}\mathcal{H}^r$ -terminology and $\Sigma = \{A, B\}$ be a signature. It holds that both sets, $\mathcal{M}_1 = \{A \sqsubseteq X \sqcap Y, X \sqsubseteq B\}$ and $\mathcal{M}_2 = \{A \sqsubseteq X \sqcap Y, Y \sqsubseteq Z, Z \sqsubseteq B\}$, are minimal basic $\mathcal{EL}\mathcal{H}^r$ -subsumption modules of \mathcal{T} w.r.t. Σ , whereas \mathcal{M}_1 is the smallest minimal basic $\mathcal{EL}\mathcal{H}^r$ -subsumption module of \mathcal{T} w.r.t. Σ as $|\mathcal{M}_1| < |\mathcal{M}_2|$.*

The notion of a justification for a concept inclusion α has been introduced as a minimal subset of a TBox that entails a given concept inclusion [4]. We can understand a minimal module as a more general notion of justification: a minimal basic $\mathcal{EL}\mathcal{H}^r$ -subsumption module of \mathcal{T} w.r.t. Σ is a justification for all the concept inclusions over Σ entailed by \mathcal{T} .

Semantic modules of $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies that are self-contained or depleting (in fact, such modules have both properties [27]) can be strictly larger than basic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -subsumption modules as illustrated by the following example.

Example 97 Let $\mathcal{T} = \{A \sqsubseteq \exists r.B\}$ be an $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminology, and $\Sigma = \{A, B\}$ be a signature. It is easy to verify that \mathcal{T} itself is a basic, self-contained, and depleting semantic module of \mathcal{T} w.r.t. Σ [25, 27], whereas the empty set is the minimal basic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -subsumption module of \mathcal{T} w.r.t. Σ .

The following example extends Example 96 to show that the modules computed by the system MEX [25] as well as the modules based on syntactic locality can be strictly larger than minimal basic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -subsumption modules as introduced in Definition 93. Note that MEX-modules are semantic modules that are self-contained as well as depleting [27] (equivalently, weak and strong [25]).

Example 98 Consider an $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminology $\mathcal{T} = \{A \sqsubseteq X \sqcap Y \sqcap U, X \sqsubseteq B, Y \sqsubseteq Z, Z \sqsubseteq B, U \equiv V \sqcap W\}$ and a signature and $\Sigma = \{A, B\}$. It holds that both sets, $\mathcal{M}_1 = \{A \sqsubseteq X \sqcap Y, X \sqsubseteq B\}$ and $\mathcal{M}_2 = \{A \sqsubseteq X \sqcap Y, Y \sqsubseteq Z, Z \sqsubseteq B\}$, are minimal basic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -subsumption modules of \mathcal{T} w.r.t. Σ . Moreover, MEX outputs $\mathcal{M}_3 = \mathcal{M}_1 \cup \mathcal{M}_2 = \{A \sqsubseteq X \sqcap Y \sqcap U, X \sqsubseteq B, Y \sqsubseteq Z, Z \sqsubseteq B\}$ as module of \mathcal{T} w.r.t. Σ . Finally, \mathcal{T} itself is the $\perp\top^*$ -local module of \mathcal{T} w.r.t. Σ .

In general, there can be several minimal basic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -subsumption modules of an acyclic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminology for a signature, and even the smallest of such modules are not necessarily unique. The next example shows a sequence of acyclic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies whose number of minimal basic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -subsumption modules for a given signature is exponentially increasing.

Example 99 Let $\mathcal{T}_n = \{A \sqsubseteq X_0\} \cup \{X_{i-1} \sqsubseteq Y_i \sqcap Z_i \mid 1 \leq i \leq n\} \cup \{Y_i \sqsubseteq X_i, Z_i \sqsubseteq X_i \mid 1 \leq i \leq n\} \cup \{X_n \sqsubseteq B\}$ with $n \geq 0$ be $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies, and let $\Sigma = \{A, B\}$ be a signature.

It holds that the set $\{A \sqsubseteq X_0, X_0 \sqsubseteq B\}$ is the minimal basic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -subsumption module of \mathcal{T}_0 w.r.t. Σ . The sets $\{A \sqsubseteq X_0, X_0 \sqsubseteq Y_1 \sqcap Z_1, Y_1 \sqsubseteq X_1, X_1 \sqsubseteq B\}$ and $\{A \sqsubseteq X_0, X_0 \sqsubseteq Y_1 \sqcap Z_1, Z_1 \sqsubseteq X_1, X_1 \sqsubseteq B\}$ are the two minimal basic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -subsumption modules of \mathcal{T}_1 . Moreover, the four sets $\{A \sqsubseteq X_0, X_0 \sqsubseteq Y_1 \sqcap Z_1, Y_1 \sqsubseteq X_1, X_1 \sqsubseteq Y_2 \sqcap Z_2, Y_2 \sqsubseteq X_2, X_2 \sqsubseteq B\}$ and $\{A \sqsubseteq X_0, X_0 \sqsubseteq Y_1 \sqcap Z_1, Y_1 \sqsubseteq X_1, X_1 \sqsubseteq Y_2 \sqcap Z_2, Z_2 \sqsubseteq X_2, X_2 \sqsubseteq B\}$ as well as $\{A \sqsubseteq X_0, X_0 \sqsubseteq Y_1 \sqcap Z_1, Z_1 \sqsubseteq X_1, X_1 \sqsubseteq Y_2 \sqcap Z_2, Y_2 \sqsubseteq X_2, X_2 \sqsubseteq B\}$ and $\{A \sqsubseteq X_0, X_0 \sqsubseteq Y_1 \sqcap Z_1, Z_1 \sqsubseteq X_1, X_1 \sqsubseteq Y_2 \sqcap Z_2, Z_2 \sqsubseteq X_2, X_2 \sqsubseteq B\}$ are the minimal basic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -subsumption modules of \mathcal{T}_2 , etc. In general, it can readily be verified that \mathcal{T}_n has 2^n many distinct minimal basic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -subsumption modules w.r.t. Σ .

In the remainder of this section, we present two different approaches for computing minimal basic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -subsumption modules. We will simply write *module* instead of ‘basic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -subsumption module’ in the remainder of this chapter.

4.2 Black-Box Algorithm

In this section, we extend the black-box approach for finding one or all justifications in [20], which is based on Reiter’s hitting set algorithm [38]. Instead of ensuring that a given entailment is preserved, we introduce an oracle to determine the inseparability between the original ontology and the resulting module. As an oracle we use a variant of the system CEX, which is a tool for deciding whether two \mathcal{ELH}^r -terminologies are logically different w.r.t. a signature [28, 24, 34]. Additionally, several optimisations to speed up the computation of minimal modules are investigated. We present an experimental evaluation of our algorithms by applying them on the prominent and large medical ontologies Snomed CT and NCI. We note that our algorithms are applicable to ontologies formulated in any ontology language provided that a tool is available that can effectively decide the inseparability relation. As CEX works with variants of \mathcal{ELH}^r -terminologies, we restrict the presentation of our algorithms to \mathcal{ELH}^r -terminologies.

4.2.1 Computing a Single Minimal Module

A first straightforward procedure SINGLE-MINIMAL-MODULE for computing a minimal module of an \mathcal{ELH}^r -terminology \mathcal{T} w.r.t. a signature Σ is given in Algorithm 13.¹ The procedure operates as follows. First, the variable \mathcal{M} is initialised with \mathcal{T} . Subsequently, the procedure iterates over every axiom $\alpha \in \mathcal{T}$ and checks whether $\text{cDiff}(\mathcal{T}, \mathcal{M} \setminus \{\alpha\}) = \emptyset$, in which case the axiom α is removed from \mathcal{M} . During the execution of the while-loop the set \mathcal{M} is hence shrunk by removing axioms that do not lead to a logical difference until a minimal module of \mathcal{T} for Σ remains.

Algorithm 13: Computing a Single Minimal Module w.r.t. a Signature

```

1 function SINGLE-MINIMAL-MODULE ( $\mathcal{T}, \Sigma$ )
2    $\mathcal{M} := \mathcal{T}$ 
3   for every axiom  $\alpha \in \mathcal{T}$  do
4     if  $\text{cDiff}(\mathcal{T}, \mathcal{M} \setminus \{\alpha\}) = \emptyset$  then
5        $\mathcal{M} := \mathcal{M} \setminus \{\alpha\}$ 
6     end
7   end
8   return  $\mathcal{M}$ 

```

Note that the minimal module that is extracted by Algorithm 13 depends on the order in which axioms were chosen during the iteration (Line 3), i.e. by iterating over the axioms in a different order one can potentially obtain a different minimal module. Moreover, one can show that all minimal modules can be computed by using all possible orderings on the axioms $\alpha \in \mathcal{T}$ in the for-loop in Line 3.

¹A similar algorithm for DL-Lite ontologies has already been described in Theorem 67 of [31].

It is easy to see that Algorithm 13 always terminates and that it runs in polynomial time in the size of \mathcal{T} and Σ since deciding the existence of a logical difference between \mathcal{ELH}^r -terminologies can be performed in polynomial time in the size of \mathcal{T} and Σ .

Regarding correctness, if we assume towards a contradiction that a set $\mathcal{M}_{\min} \subseteq \mathcal{T}$ computed by Algorithm 13 applied on \mathcal{T} and Σ is not a minimal module of \mathcal{T} w.r.t. Σ , then there would exist an axiom $\alpha \in \mathcal{M}$ such that $\text{cDiff}(\mathcal{T}, \mathcal{M}_{\min} \setminus \{\alpha\}) = \emptyset$. However, when α was analysed in the for-loop in Line 3, $\text{cDiff}(\mathcal{T}, \mathcal{M}' \setminus \{\alpha\})$ must have been empty as well by monotonicity of \models , where \mathcal{M}' with $\mathcal{M}_{\min} \subseteq \mathcal{M}'$ represents the value of the variable \mathcal{M} in Algorithm 13 at the time α was inspected. Consequently, it would hold that $\alpha \notin \mathcal{M}_{\min}$ and we have derived a contradiction. We hence obtain the following result.

Theorem 100 *Let \mathcal{T} be an \mathcal{ELH}^r -terminology and let Σ be a signature. Then Algorithm 13 applied on \mathcal{T} and Σ computes a minimal module of \mathcal{T} for Σ .*

Proof Let \mathcal{M} be the module of \mathcal{T} returned by Algorithm 13 applied on (\mathcal{T}, Σ) .

Furthermore, let $\mathcal{M}_0, \dots, \mathcal{M}_n$ be the sequence of modules of \mathcal{T} computed by Algorithm 13 such that $\mathcal{M}_0 = \mathcal{T}$ and $\mathcal{M}_n = \mathcal{M}$.

If we assume towards a contradiction that \mathcal{M} is not minimal, then there exists an axiom $\alpha \in \mathcal{M}$ such that $\text{cDiff}(\mathcal{T}, \mathcal{M} \setminus \{\alpha\}) = \emptyset$. Hence, there exists $0 \leq i \leq n$ such that $\text{cDiff}_{\Sigma}(\mathcal{T}, \mathcal{M}_i \setminus \{\alpha\}) \neq \emptyset$. As $\mathcal{M}_n \subseteq \mathcal{M}_i$, we can infer that $\mathcal{M}_n \setminus \{\alpha\} \subseteq \mathcal{M}_i \setminus \{\alpha\}$ and $\text{cDiff}_{\Sigma}(\mathcal{T}, \mathcal{M}_i \setminus \{\alpha\}) \subseteq \text{cDiff}_{\Sigma}(\mathcal{T}, \mathcal{M}_n \setminus \{\alpha\})$ by Lemma 27. Thus, $\text{cDiff}_{\Sigma}(\mathcal{T}, \mathcal{M}_n \setminus \{\alpha\}) \neq \emptyset$, contradicting our assumption. \square

As checking the existence of a logical difference can be costly in practice, we now introduce a refinement of the previous algorithm that potentially allows it to reduce the number of logical difference checks that are required for computing a minimal module. The refined procedure SINGLE-MINIMAL-MODULE-BUBBLE is shown in Algorithm 14.

Intuitively, instead of checking whether the removal of a single axiom leads to a logical difference, the refined procedure removes a set \mathcal{B} of axioms from \mathcal{T} at once. Such a set \mathcal{B} is also called a *bubble*. As an additional optimisation we introduce the notion of logical difference core, which will become relevant in the context of computing all minimal modules when the algorithm for computing one minimal module has to be executed frequently.

Definition 101 (Logical Difference Core) *Let \mathcal{T} be an \mathcal{ELH}^r -terminology and let Σ be a signature. A subset $\mathcal{C} \subseteq \mathcal{T}$ is said to be a logical difference core of \mathcal{T} w.r.t. Σ iff for every $\alpha \in \mathcal{C}$ it holds that $\text{cDiff}(\mathcal{T}, \mathcal{T} \setminus \{\alpha\}) \neq \emptyset$.*

Given a logical difference core \mathcal{C} of \mathcal{T} w.r.t. Σ and a minimal module \mathcal{M} of \mathcal{T} w.r.t. Σ , it is easy to see that $\mathcal{C} \subseteq \mathcal{M}$ must hold. The maximal logical difference core can be computed by collecting all the axioms $\alpha \in \mathcal{T}$ for which $\text{cDiff}(\mathcal{T}, \mathcal{T} \setminus \{\alpha\}) \neq \emptyset$.

Algorithm 14: Computing a Single Minimal Module w.r.t. a Signature using Axiom Bubbles

```

1 function SINGLE-MINIMAL-MODULE-BUBBLE ( $\mathcal{T}, \Sigma, n, \mathcal{C}$ )
2    $\mathcal{M} := \mathcal{T}$ 
3    $\mathcal{Q} := \text{SPLIT}(\mathcal{T} \setminus \mathcal{C}, n)$ 
4   while  $\mathcal{Q} \neq []$  do
5      $\mathcal{B} := \text{HEAD}(\mathcal{Q})$ 
6      $\mathcal{Q} := \text{TAIL}(\mathcal{Q})$ 
7     if  $\text{cDiff}(\mathcal{T}, \mathcal{M} \setminus \mathcal{B}) = \emptyset$  then
8        $\mathcal{M} := \mathcal{M} \setminus \mathcal{B}$ 
9     end
10    else
11      if  $|\mathcal{B}| > 1$  then
12         $(\mathcal{B}_l, \mathcal{B}_r) := \text{SPLITHALF}(\mathcal{B})$ 
13         $\mathcal{Q} := \mathcal{B}_l :: \mathcal{B}_r :: \mathcal{Q}$ 
14      end
15    end
16  end
17  return  $\mathcal{M}$ 

```

Lemma 102 Let \mathcal{T} be an \mathcal{ELH}^r -terminology and let Σ be a signature. Additionally, let $\mathcal{C} \subseteq \mathcal{T}$ be a logical difference core of \mathcal{T} w.r.t. Σ and \mathbb{M} is the set of all minimal modules of \mathcal{T} w.r.t. Σ . Then

$$\mathcal{C} = \bigcap_{\mathcal{M} \in \mathbb{M}} \mathcal{M}.$$

Proof Let $\mathcal{S} = \bigcap_{\mathcal{M} \in \mathbb{M}} \mathcal{M}$. First, we prove that $\mathcal{S} \subseteq \mathcal{C}$. For every axiom $\alpha \in \mathcal{S}$, we have $\alpha \in \bigcap_{\mathcal{M} \in \mathbb{M}} \mathcal{M}$. As \mathcal{M} is minimal w.r.t. set inclusion, we could get that $\text{cDiff}_{\Sigma}\{\mathcal{T}, \mathcal{T} \setminus \{\alpha\}\} \neq \emptyset$. By Definition 101, we could conclude that for every axiom, $\alpha \in \mathcal{C}$. Hence, $\mathcal{S} \subseteq \mathcal{C}$.

Then we prove that $\mathcal{C} \subseteq \mathcal{S}$. We assume now towards a contradiction that there exists an axiom α such that $\alpha \in \mathcal{C}$, but $\alpha \notin \mathcal{S}$. As $\alpha \notin \mathcal{S}$, there exist a minimal module \mathcal{M} such that $\alpha \notin \mathcal{M}$ and $\text{cDiff}_{\Sigma}(\mathcal{T}, \mathcal{M}) = \emptyset$. As $\mathcal{M} \subseteq \mathcal{T} \setminus \{\alpha\}$. By Lemma 27, we have that $\text{cDiff}_{\Sigma}(\mathcal{T}, \mathcal{T} \setminus \{\alpha\}) = \emptyset$, which contradicts to our assumption that $\alpha \in \mathcal{C}$, which implies that $\text{cDiff}_{\Sigma}(\mathcal{T}, \mathcal{T} \setminus \{\alpha\}) \neq \emptyset$. \square

Lemma 102 shows that every axiom in logical difference core is contained in every minimal module. Therefore, the black-box algorithm does not need to check the axioms in the logical different core.

Now, the procedure SINGLE-MINIMAL-MODULE-BUBBLE applied on a terminology \mathcal{T} , a signature Σ , an initial size parameter n for the bubbles, and a logical difference core \mathcal{C} of \mathcal{T} w.r.t. Σ operates as follows. First, the variable \mathcal{M} is set to contain all the axioms of \mathcal{T} and the bubble queue \mathcal{Q}

is initialised by partitioning the axioms contained in $\mathcal{T} \setminus \mathcal{C}$ into bubbles of size n . Note that the size of one bubble may be different from n if n is not a divisor of $|\mathcal{T}|$, or if $n > |\mathcal{T}|$. The resulting bubbles are then stored in the queue \mathbb{Q} . As long as \mathbb{Q} is not empty, the first bubble \mathcal{B} is extracted from the queue (lines 5 and 6). Note that the empty queue is denoted with $[\]$. Subsequently, it is verified in Line 7 whether the removal of the axioms in \mathcal{B} from the minimal module candidate \mathcal{M} leads to a logical difference. If not, all the axioms in \mathcal{B} can safely be removed from \mathcal{M} in Line 8. Otherwise, if the bubble contained more than one axiom (Line 10), we have to identify the subsets of \mathcal{B} whose removal does not yield a logical difference. To that end, \mathcal{B} is split into two bubbles \mathcal{B}_l and \mathcal{B}_r (Line 11) such that $\mathcal{B}_l, \mathcal{B}_r \subseteq \mathcal{B}$, $|\mathcal{B}_l| = \lceil \frac{1}{2} \cdot |\mathcal{B}| \rceil$, and $|\mathcal{B}_r| = \lfloor \frac{1}{2} \cdot |\mathcal{B}| \rfloor$. The bubbles \mathcal{B}_l and \mathcal{B}_r are then prepended to the queue (Line 12), and the algorithm continues with the next iteration.

The correctness of Algorithm 14 can be shown as before with Algorithm 13. Termination on any input follows from the fact that every axiom in \mathcal{T} appears in at most one bubble in \mathbb{Q} and that in each iteration either the overall number of bubbles is reduced, or one bubble that contains more than one axiom is split into two smaller bubbles. Note that once a bubble \mathcal{B} of size 1 has been selected in Line 5, it will not be contained in \mathbb{Q} in subsequent iterations. We obtain the following result.

Theorem 103 *Let \mathcal{T} be an \mathcal{ELH}^r -terminology and let Σ be a signature. Additionally, let $\mathcal{C} \subseteq \mathcal{T}$ be a logical difference core of \mathcal{T} w.r.t. Σ . Then Algorithm 14 applied on \mathcal{T} , Σ , and \mathcal{C} computes a minimal module of \mathcal{T} for Σ .*

Regarding computational complexity, we observe that the decomposition of every bubble \mathcal{B} induces a binary tree in which the nodes are labelled with the bubbles resulting from splitting the parent bubble. In our algorithm, given a bubble \mathcal{B} , such a decomposition tree has a depth of at most $\lceil \log_2 |\mathcal{B}| \rceil$ and the number of nodes in a decomposition tree corresponds to the number of logical difference checks. As the number of nodes in a binary tree of depth h is bounded by $2^{h+1} - 1$, we hence obtain that every initial bubble \mathcal{B} results in at most $2 \cdot |\mathcal{B}| - 1$ logical difference checks. Overall, we can infer that the procedure SINGLE-MINIMAL-MODULE-BUBBLE runs in polynomial time in the size of \mathcal{T} , Σ , and n .

4.2.2 Computing All Minimal Modules

A naïve way to compute all minimal modules is to enumerate all subsets of the input TBox \mathcal{T} and to check their logical difference w.r.t. \mathcal{T} and a given signature. For \mathcal{ELH}^r -terminologies the logical difference problem can be decided in polynomial time [23]. Example 99 shows that there are \mathcal{ELH}^r -terminologies with exponentially many minimal modules. Consequently, computing all minimal modules of an \mathcal{ELH}^r -terminology can only be achieved in time exponential in the size of the terminology in the worst case.

For that reason, upper approximations of (the union of) all minimal modules such as the syntactic locality-based module notions that can be extracted more efficiently have been introduced [16]. In our algorithm for computing all minimal modules (and in our experiments for extracting one minimal

module) we will make use of syntactic $\perp\top^*$ -locality modules to speed up computations. These modules are among the smallest modules based on syntactic locality notions [42]. They can be obtained by iterating the process of extracting a syntactic \perp -local module followed by extracting a syntactic \top -local module until a fixpoint is reached. We will extract syntactic $\perp\top^*$ -locality modules using the OWLAPI. Note that any syntactic $\perp\top^*$ -locality module \mathcal{M}_s of an \mathcal{ELH}^r -terminology \mathcal{T} w.r.t. a signature Σ contains all the minimal modules of \mathcal{T} w.r.t. Σ .

In our algorithm for computing all minimal modules, we make use of a technique developed for computing all minimal hitting sets [38]. Our algorithm is based on the following observation: given a minimal module \mathcal{M} of \mathcal{T} w.r.t. a signature Σ , then for any other minimal module \mathcal{M}' of \mathcal{T} w.r.t. Σ there must exist $\alpha \in \mathcal{M}'$ such that $\alpha \notin \mathcal{M}$, i.e. \mathcal{M}' must be contained in $\mathcal{T} \setminus \{\alpha\}$ for some $\alpha \in \mathcal{M}$.

Similarly to [38], our algorithm organises the search space using a labelled, directed tree τ , called *module search tree for \mathcal{T}* , that is extended during the run of the algorithm. Formally, τ is a tuple $(\mathcal{V}, \mathcal{E}, \mathcal{L}, \rho)$, where \mathcal{V} is a non-empty, finite set of nodes, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is a set of edges, \mathcal{L} is an edge labelling function, mapping every edge $e \in \mathcal{E}$ to an axiom $\alpha \in \mathcal{T}$, and $\rho \in \mathcal{V}$ is the root node of τ . The procedure ALL-MINIMAL-MODULES shown in Algorithm 15 operates on a queue \mathbb{Q} that contains the nodes of τ that still have to be expanded. Intuitively, the labels of the edges on the unique path from the root node to a node $v \in \mathcal{V}$ are the axioms that should be excluded from the search for minimal modules. In each iteration a node v is extracted from \mathbb{Q} and the set $\mathcal{T}_{\text{ex}} \subseteq \mathcal{T}$ of exclusion axioms is computed by analysing the path from the root node to v . The procedure SINGLE-MINIMAL-MODULE-BUBBLE is then used to find a minimal module \mathcal{M} of $\mathcal{T} \setminus \mathcal{T}_{\text{ex}}$ w.r.t. Σ . Subsequently, the tree τ is extended by adding a child v_α of v for every $\alpha \in \mathcal{M}$ and the search for all minimal modules continues in the next iteration on the newly added nodes v_α .

We now describe the ALL-MINIMAL-MODULES procedure in detail, together with the optimisations that we implemented. Some of the improvements to prune the search space have been proposed in [38] already.

Given an \mathcal{ELH}^r -terminology \mathcal{T} , a signature Σ , a bubble size $n \geq 1$, and a logical difference core $\mathcal{C} \subseteq \mathcal{T}$ of \mathcal{T} w.r.t. Σ as input, in the lines 2 and 3 a syntactic $\perp\top^*$ -locality module \mathcal{T}_Σ of \mathcal{T} w.r.t. Σ is extracted from \mathcal{T} , the variable τ is initialised to represent a module search tree for \mathcal{T} having only one node ρ . Moreover, the variables $\mathbb{M} \subseteq 2^{\mathcal{T}_\Sigma}$, containing the minimal modules that have been computed so far, and $\mathbb{W} \subseteq \mathcal{V}$, containing the already explored nodes of τ , are both initialised with the empty set. The queue \mathbb{Q} of nodes in τ that still have to be explored is also set to contain the node ρ as its only element.

The algorithm then enters a while-loop in the lines 4 to 29 in which it remains as long as \mathbb{Q} is not empty. In each iteration the first element v is extracted from \mathbb{Q} and v is added to \mathbb{W} (lines 5 to 7). Subsequently, the axioms labelling the edges of the path π_v from ρ to v in τ are collected in the set \mathcal{T}_{ex} (Line 8). The algorithm then checks whether π_v is redundant, in which case the next iteration of the while-loop starts.

The path π_v is redundant iff there exists an already explored node $w \in \mathbb{W}$ such that (a) the axioms in \mathcal{T}_{ex} are exactly the axioms labelling the edges of the path π_w from ρ to w in τ , or (b) w is a leaf node

Algorithm 15: Computing All Minimal Modules w.r.t. a Signature

```

1 function ALL-MINIMAL-MODULES ( $\mathcal{T}, \Sigma, n, \mathcal{C}$ )
2    $\mathcal{T}_\Sigma := \text{SYNTACTICLOCALITYMODULE}(\mathcal{T}, \Sigma)$ 
3    $\mathbb{M} := \emptyset; \tau := (\{\rho\}, \emptyset, \emptyset, \rho); \mathbb{Q} := [\rho]; \mathbb{W} := \emptyset$ 
4   while  $\mathbb{Q} \neq []$  do
5      $v := \text{HEAD}(\mathbb{Q})$ 
6      $\mathbb{Q} := \text{TAIL}(\mathbb{Q})$ 
7      $\mathbb{W} := \mathbb{W} \cup \{v\}$ 
8      $\mathcal{T}_{ex} := \text{LABELS}(\text{PATH}(\tau, \rho, v))$ 
9     if IS-PATH-REDUNDANT( $\tau, \rho, \mathcal{T}_{ex}, \mathbb{W}$ ) then
10      | continue
11    end
12    if  $c\text{Diff}(\mathcal{T}_\Sigma, \mathcal{T}_\Sigma \setminus \mathcal{T}_{ex}) \neq \emptyset$  then
13      | continue
14    end
15     $\mathcal{M} := \emptyset$ 
16    if there exists  $\mathcal{M}' \in \mathbb{M}$  such that  $\mathcal{T}_{ex} \cap \mathcal{M}' = \emptyset$  then
17      |  $\mathcal{M} := \mathcal{M}'$ 
18    end
19    else
20      |  $\mathcal{M} := \text{SINGLE-MINIMAL-MODULE-BUBBLE}(\mathcal{T}_\Sigma \setminus \mathcal{T}_{ex}, \Sigma, n, \mathcal{C})$ 
21      | if  $\mathcal{M} = \mathcal{C}$  then
22        | return  $\{\mathcal{C}\}$ 
23      | end
24      |  $\mathbb{M} := \mathbb{M} \cup \{\mathcal{M}\}$ 
25    end
26    for every  $\alpha \in \mathcal{M} \setminus \mathcal{C}$  do
27      |  $v_\alpha := \text{ADDCHILD}(\tau, v, \alpha)$ 
28      |  $\mathbb{Q} := v_\alpha :: \mathbb{Q}$ 
29    end
30  end
31  return  $\mathbb{M}$ 

```

of τ and the edges of π_w are only labelled with axioms from \mathcal{T}_{ex} . Condition (a) corresponds to *early path termination* in [38, 20]: the existence of π_w implies that all possible extensions of π_v have already been considered. Condition (b) implies that the axioms labelling the edges of π_w lead to a logical difference when removed from \mathcal{T}_Σ . Consequently, removing \mathcal{T}_{ex} from \mathcal{T}_Σ also induces a logical difference by monotonicity of \models , implying that π_v and all its extensions do not have to be explored. Moreover, the current iteration can also be terminated immediately if $\text{cDiff}(\mathcal{T}_\Sigma, \mathcal{T}_\Sigma \setminus \mathcal{T}_{\text{ex}}) \neq \emptyset$ (lines 12 to 14) as no subset of $\mathcal{T}_\Sigma \setminus \mathcal{T}_{\text{ex}}$ can be a module of \mathcal{T}_Σ (and therefore of \mathcal{T}) w.r.t. Σ .

Subsequently, in Line 15 the variable \mathcal{M} that will hold a minimal module of $\mathcal{T}_\Sigma \setminus \mathcal{T}_{\text{ex}}$ is initialised with \emptyset . At this point we can check if a minimal module $\mathcal{M}' \in \mathbb{M}$ has already been computed for which $\mathcal{T}_{\text{ex}} \cap \mathcal{M}' = \emptyset$ (lines 16 and 17) holds, in which case we set \mathcal{M} to \mathcal{M}' . This optimisation step can also be found in [38, 20] and it allows us to avoid a costly call to the SINGLE-MINIMAL-MODULE-BUBBLE procedure. Otherwise, in the lines 18 to 24 we have to apply SINGLE-MINIMAL-MODULE-BUBBLE on $\mathcal{T}_\Sigma \setminus \mathcal{T}_{\text{ex}}$ to obtain a minimal module of $\mathcal{T}_\Sigma \setminus \mathcal{T}_{\text{ex}}$ w.r.t. Σ . The algorithm then checks whether \mathcal{M} is equal to \mathcal{C} (lines 20 to 22), in which case the search for additional modules can be aborted. If the logical difference core \mathcal{C} is a minimal module itself, we can infer that no other minimal module exists since \mathcal{C} is a subset of all the minimal modules. Otherwise, the module \mathcal{M} is added to \mathbb{M} in Line 23. Finally, in the lines 25 to 28 the tree τ is extended by adding a child v_α to v for every $\alpha \in \mathcal{M} \setminus \mathcal{C}$, connected by an edge labelled with α . Note that it is sufficient to take $\alpha \notin \mathcal{C}$ as a set \mathcal{M} with $\mathcal{C} \not\subseteq \mathcal{M}$ cannot be a minimal module of \mathcal{T} w.r.t. Σ . The procedure finishes by returning the set \mathbb{M} in Line 30.

Regarding correctness of Algorithm 15, we note that only minimal modules are added to \mathbb{M} . For completeness, one can show that the locality-based module \mathcal{T}_Σ of \mathcal{T} w.r.t. Σ contains all the minimal modules of \mathcal{T} w.r.t. Σ . Moreover, it is easy to see that the proposed optimisations do not lead to a minimal module not being computed. Overall, we obtain the following result.

Theorem 104 *Let \mathcal{T} be an \mathcal{ELH}^r -terminology and let Σ be a signature. Additionally, let $n \geq 1$, and let $\mathcal{C} \subseteq \mathcal{T}$ be a logical difference core of \mathcal{T} w.r.t. Σ .*

Then the procedure ALL-MINIMAL-MODULES shown in Algorithm 15 and applied on \mathcal{T} , Σ , n , and \mathcal{C} , exactly computes all the minimal modules of \mathcal{T} for Σ .

Algorithm 15 terminates on any input as the paths in the module search tree τ for \mathcal{T} that is constructed during the execution represent all the permutations of the axioms in \mathcal{T} that are relevant for finding all minimal modules. It is easy to see that the procedure ALL-MINIMAL-MODULES runs in exponential time in size of \mathcal{T} (and polynomially in Σ , n , and \mathcal{C}) in the worst case.

The following proposition shows the situation that there exists only one minimal module, which can be considered as an approximation of Algorithm 15.

Proposition 105 *Let \mathcal{T} be an \mathcal{ELH}^r -terminology and Σ a signature. Additionally, let $\mathcal{C} \subseteq \mathcal{T}$ be a logical difference core of \mathcal{T} w.r.t. Σ , \mathbb{M} be the set of all minimal modules of \mathcal{T} w.r.t. Σ and $\mathcal{M} \in \mathbb{M}$. It holds that $\mathcal{C} = \mathcal{M}$ iff $|\mathbb{M}| = 1$.*

4.3 Glass-Box Algorithm

An alternative to the black-box approach for computing minimal modules is to directly select the relevant axioms from an ontology that preserve the desired entailments, which can be done by combining subsumption justification of every concept and role name in signature. In this section, we introduce a glass-box algorithm based on computing subsumption justifications that we introduce in Chapter 3.

The following theorems show how to use subsumption justifications to compute all minimal modules for \mathcal{ELH} -terminology (Theorem 106) and \mathcal{ELH}^r -terminology (Theorem 107).

Theorem 106 *Let \mathcal{T} be an acyclic and normalised \mathcal{ELH} -terminology, Σ be a signature and $A \in \Sigma \cap N_C$. Additional, let $\mathbb{J}_{\mathcal{T}}^{\rightarrow}(A, \Sigma)$ be the set of all $\langle A, \Sigma \rangle$ -subsumer justifications of \mathcal{T} w.r.t. Σ . Similarly, let $\mathbb{J}_{\mathcal{T}}^{\leftarrow}(A, \Sigma)$ be the set of all $\langle A, \Sigma \rangle$ -subsumee justifications of \mathcal{T} w.r.t. Σ and $\mathbb{J}_{\mathcal{T}}^{\leftarrow}(r, \Sigma)$ be the set of all $\langle r, \Sigma \rangle$ -role subsumption justifications of \mathcal{T} w.r.t. Σ . Then the set of all minimal basic \mathcal{ELH} -modules of \mathcal{T} w.r.t. Σ is*

$$\mathbb{M}_{\Sigma}^{\mathcal{T}} := \text{Minimize}_{\subseteq} \left(\bigotimes_{r \in \Sigma \cap N_R} \mathbb{J}_{\mathcal{T}}(r, \Sigma) \otimes \bigotimes_{A \in \Sigma \cap N_C} \mathbb{J}_{\mathcal{T}}^{\rightarrow}(A, \Sigma) \otimes \bigotimes_{A \in \Sigma \cap N_C} \mathbb{J}_{\mathcal{T}}^{\leftarrow}(A, \Sigma) \right).$$

Theorem 106 can be proved by Theorem 24 and Proposition 57. The idea of the proof is that we need one subsumption justification, for every potential difference witness (cf. Theorem 24), to be contained in a minimal module in order to prevent the witness (cf. Proposition 57).

As the concept $\text{dom}(r)$ and $\text{ran}(r)$, $r \in \Sigma$ can also be primitive witness on the left-hand side for \mathcal{ELH}^r terminology, so compared with \mathcal{ELH} terminology, additionally we need to combine all $\langle \varphi', \Sigma \rangle$ -subsumer justifications, $\varphi' \in \{\text{dom}(t), \text{ran}(t) \mid t \in N_R \cap \Sigma\}$ for computing all minimal modules for \mathcal{ELH}^r -terminologies.

Theorem 107 *Let \mathcal{T} be an acyclic and normalised \mathcal{ELH}^r -terminology, Σ be a signature and $A \in \Sigma \cap N_C$. Additional, let $\mathbb{J}_{\mathcal{T}}^{\rightarrow}(\varphi, \Sigma)$ be the set of all $\langle \varphi, \Sigma \rangle$ -subsumer justifications of \mathcal{T} w.r.t. Σ and $\varphi \in (\Sigma \cap N_C) \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}$. Similarly, let $\mathbb{J}_{\mathcal{T}}^{\leftarrow}(A, \Sigma)$ be the set of all $\langle A, \Sigma \rangle$ -subsumee justifications of \mathcal{T} w.r.t. Σ . where $\Sigma^{\text{dom}} = \{\text{dom}(t) \mid t \in N_R \cap \Sigma\}$ and $\Sigma^{\text{ran}} = \{\text{ran}(t) \mid t \in N_R \cap \Sigma\}$. Then the set of all minimal basic \mathcal{ELH}^r -modules of \mathcal{T} w.r.t. Σ is*

$$\mathbb{M}_{\Sigma}^{\mathcal{T}} := \text{Minimize}_{\subseteq} \left(\bigotimes_{r \in \Sigma \cap N_R} \mathbb{J}_{\mathcal{T}}(r, \Sigma) \otimes \bigotimes_{\varphi \in (\Sigma \cap N_C) \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}} \mathbb{J}_{\mathcal{T}}^{\rightarrow}(\varphi, \Sigma) \otimes \bigotimes_{A \in \Sigma \cap N_C} \mathbb{J}_{\mathcal{T}}^{\leftarrow}(A, \Sigma) \right).$$

Similar to Theorem 106, Theorem 107 can be proved by Theorem 24 and Proposition 82.

4.4 Evaluation

To demonstrate the practical applicability of our approach, we have implemented Algorithms 14 and 15 in a Java prototype to compute one and all minimal basic \mathcal{ELH}^r -subsumption modules of

biomedical ontologies: Snomed CT (version Jan 2016), an acyclic \mathcal{EL} -terminology consisting of 317 891 axioms, and NCIIt (version 18.04e), a cyclic \mathcal{ELH}^r -terminology containing 139 748 axioms. The experiments have been carried out on machines equipped with an Intel Xeon Core 4 Duo CPU running at 2.50GHz and with 64GiB of RAM.

Black-box Algorithm on \mathcal{EL} - and \mathcal{ELH}^r -Terminology

Tables 4.1 and 4.2 show the results for computing one minimal basic \mathcal{ELH}^r -subsumption module of Snomed CT and NCIIt for 100 random signatures of different sizes. When the size of the signature increases, it takes more time in general to compute one minimal module and the size of their minimal module is also increasing. Moreover, in our experiments the median computation times were decreasing with an increasing bubble size for signatures with 200 concept names.

Table 4.3 shows that there exist several minimal basic subsumption modules of Snomed CT for the selected signatures (which contain concept names connected to at most 8 other axioms). Similarly, Table 4.4 also shows that there also exist several minimal basic subsumption modules of NCIIt for the random signatures. In our experiments the number of minimal modules rose up to 32, and the size of the minimal modules varied from one signature to another.

Although a precomputation of the maximal logical difference core has the potential of narrowing down the search space, it requires extra computational effort, which can be potentially very time-consuming. In order to check whether the use of the logical difference core can help to speed up the process of searching for all minimal modules, we computed all the minimal modules of Snomed CT and NCIIt with and without precomputing the maximal logical difference core for the same signatures. It turns out that in our experiments the precomputation of the maximal core was beneficial to the overall performance: the overall computation process was sped up by more than three times for Snomed CT.

Comparison between Black-box and Glass-box Approaches

In the following experiment, we compare our approaches for computing all minimal basic modules using the glass-box approach and the black-box approach in terms of computation time, as depicted in Figure 4.1. The x -axis is labelled with the sizes of the input ontologies. To obtain different sized input ontologies, we used random signatures to extract their MEX-modules [27], yielding 328 sub-ontologies of sizes ranging from 14 to 2271. Our method (red squares) was generally about 10 times faster than the search-based approach (blue triangles) except for 11 small sized input ontologies. This indicates that our approach is suitable for computing all minimal basic modules, especially for large ontologies.

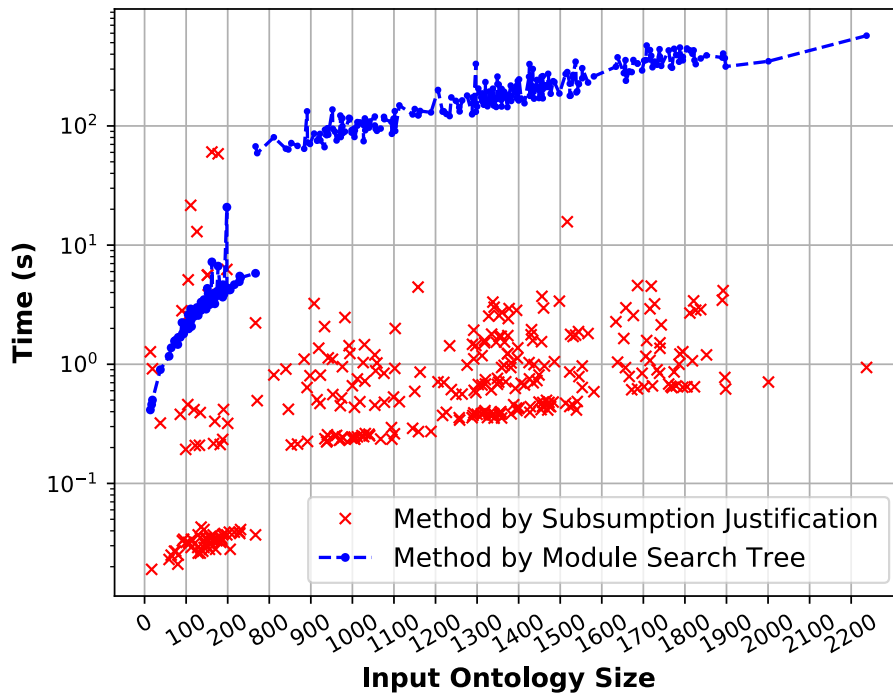


Figure 4.1 Time comparison of computing minimal modules by glass-box approach and black-box approach over different sized input ontologies

$ \Sigma \cap N_C $	50			
Bubble Size	10	25	50	100
Time (s)	67 / 523 / 200 / 87.2	68 / 483 / 197 / 82.5	70 / 505 / 202 / 85.3	71 / 507 / 204 / 86.4
Sizes	50 / 118 / 77 / 14.6	50 / 118 / 77 / 14.5	50 / 118 / 77 / 14.6	50 / 118 / 77 / 14.6
Size MEX-Mod	401 / 720 / 587 / 60.7			
Size $\perp T^*$ -Mod	1075 / 2456 / 1803 / 300.2			
$ \Sigma \cap N_C $	75			
Bubble Size	10	25	50	100
Time (s)	225 / 584 / 434 / 102.0	216 / 1359 / 531 / 209.8	226 / 575 / 447 / 101.0	231 / 586 / 450 / 101.8
Sizes	78 / 177 / 110 / 16.4	75 / 216 / 113 / 20.8	78 / 177 / 110 / 15.8	78 / 178 / 110 / 15.9
Size MEX-Mod	679 / 971 / 825 / 72.0			
Size $\perp T^*$ -Mod	1939 / 3779 / 2641 / 373.2			

Table 4.1 Computation of *one* minimal basic subsumption module of Snomed CT for 100 random signatures containing 50/75 concept names and all role names (minimal / maximal / median / standard deviation)

$ \Sigma \cap N_C $	50			
Bubble Size	10	25	50	100
Time (s)	13 / 319 / 53.5 / 93.1	13 / 318 / 55.0 / 96.5	14 / 328 / 54.5 / 101.0	13 / 325 / 64.0 / 97.29
Sizes	45 / 240 / 119.5 / 55.6	49 / 240 / 119.5 / 55.4	49 / 240 / 119.5 / 55.3	47 / 242 / 119.5 / 55.6
Size $\perp T^*$ -Mod	116 / 597 / 250.0 / 146.2			
$ \Sigma \cap N_C $	100			
Bubble Size	10	25	50	100
Time (s)	81 / 513 / 230.0 / 133.8	92 / 555 / 238.0 / 145.2	82 / 530 / 245.0 / 141.3	94 / 526 / 242.0 / 143.5
Sizes	145 / 307 / 222.0 / 53.1	145 / 307 / 222.0 / 53.2	145 / 307 / 221.0 / 53.0	145 / 308 / 224.0 / 52.5
Size $\perp T^*$ -Mod	328 / 1284 / 590.0 / 241.4			

Table 4.2 Computation of *one* minimal basic subsumption module of NCI for 100 random signatures containing 50/100 concept names and all role names (minimal / maximal / median / standard deviation)

Optimisation	Core $\mathcal{C} = \emptyset$	Core $\mathcal{C} \neq \emptyset$
Time (s)	5 / 709 / 24 / 238.7	2 / 118 / 7 / 36.0
Number of Modules	1 / 32 / 6 / 9.2	
Size of Modules	81 / 265 / 126 / 46.5	

Table 4.3 Computation of *all* minimal basic subsumption modules of Snomed CT for 20 selected signatures consisting of 30 concept names and all role names using a bubble size of 50 (min / max / med / std dev)

Optimisation	Core $\mathcal{C} = \emptyset$	Core $\mathcal{C} \neq \emptyset$
Time (s)	1 / 454 / 44 / 142.5	1 / 173 / 24 / 54.8
Number of Modules	1 / 30 / 3.5 / 8.2	
Size of Modules	8 / 73 / 52 / 10.4	

Table 4.4 Computation of *all* minimal basic subsumption modules of NCI for 20 random signatures consisting of 30 concept names and all role names using a bubble size of 10 (min / max / med / std dev)

Chapter 5

Extraction of Best Ontology Excerpts

Real-world ontologies are usually too large for users to comprehend. Therefore, in this section, we introduce a new notion called *best ontology excerpt*, a size-restricted subset of the ontology. We present an approach to computing best ontology excerpts by encoding the problem as a partial maximum satisfiability problem (Max-SAT). Then we introduce a notion of *weighted signature* and update our approach to computing best excerpts for weighted signature by encoding it as a weighted partial Max-SAT problem. Finally, we evaluate our approach on large ontologies.

5.1 Ontology Excerpts

In this chapter, we are interested in gaining more control over the size of a module in order to be able to reuse the knowledge contained in an ontology in a scenario where resources are restricted in terms of cognitive ability in human users, and time and space available in technical systems.

We start with a definition of ontology excerpts.

Definition 108 (Ontology Excerpt) *Let \mathcal{O} be an ontology and let $k > 0$ be a natural number. A k -excerpt of \mathcal{O} is a subset $\mathcal{E} \subseteq \mathcal{O}$ consisting of at most k axioms, i.e. $|\mathcal{E}| \leq k$.*

An ontology excerpt is a subset of the ontology of a certain size. However, we are interested in those excerpts that preserve (as much as possible) the meaning of or the knowledge about the symbols in a signature of interest. To quantify the meaning of an excerpt, we need some metric μ . We assume that the lower the value of μ for an excerpt is, the more meaning is preserved by the excerpt. This is made precise as follows.

Definition 109 (Incompleteness Measure) *Let \mathcal{O} be an ontology. An incompleteness measure μ is a function that maps every triple $(\mathcal{O}, \Sigma, \mathcal{E})$ consisting of an ontology \mathcal{O} , a signature Σ , and an excerpt $\mathcal{E} \subseteq \mathcal{O}$ to a non-negative real number.*

Given an incompleteness measure, we can define the best excerpt as follows.

Definition 110 (Best k -Excerpt) Let \mathcal{O} be an ontology, let Σ be a signature, and let $k > 0$ be a natural number. Additionally, let μ be an incompleteness measure. A best k -excerpt of \mathcal{O} w.r.t. Σ under μ is a k -excerpt \mathcal{E} of \mathcal{O} such that

$$\mu(\mathcal{O}, \Sigma, \mathcal{E}) = \min\{\mu(\mathcal{O}, \Sigma, \mathcal{E}') \mid \mathcal{E}' \text{ is a } k\text{-excerpt of } \mathcal{O}\}.$$

We first use as incompleteness measure μ the number $\text{cDiff}(\mathcal{O}, \Sigma, \mathcal{E})$ of \mathcal{E} - \mathcal{L} -concept inclusion difference witnesses in Σ w.r.t. \mathcal{O} and \mathcal{E} , which is formally defined as $\text{cDiff}(\mathcal{O}, \Sigma, \mathcal{E}) = |\text{Wtn}_\Sigma(\mathcal{O}, \mathcal{E})|$. An example of best excerpts under this incompleteness measure is illustrated below.

Example 111 (Ex. 25 contd.) Let \mathcal{O} consist of the following four axioms:

$$\begin{array}{ll} \alpha_1 : A \sqsubseteq B \sqcap \exists r.X & \alpha_2 : B \sqsubseteq A \\ \alpha_3 : X \equiv A \sqcap B & \alpha_4 : Y \equiv B \sqcap \exists r.(X \sqcap \exists s.A) \end{array}$$

For $\Sigma = \{A, B\}$, the values $\text{cDiff}(\mathcal{O}, \Sigma, \mathcal{E})$ for all 2-excerpts \mathcal{E} of \mathcal{O} are given in the second row of the table below.

$\{\alpha_1, \alpha_2\}$	$\{\alpha_1, \alpha_3\}$	$\{\alpha_1, \alpha_4\}$	$\{\alpha_2, \alpha_3\}$	$\{\alpha_2, \alpha_4\}$	$\{\alpha_3, \alpha_4\}$
0	2	2	2	2	2

One can thus see that $\{\alpha_1, \alpha_2\}$ is the best 2-excerpt of \mathcal{O} w.r.t. Σ under cDiff .

The following example shows that best k -excerpt is not monotonic w.r.t. k , the size of an excerpt.

Example 112 Let \mathcal{O} consist of the following four axioms:

$$\begin{array}{ll} \alpha_1 : A_1 \sqsubseteq A_2 & \alpha_2 : A_2 \sqsubseteq A_3 \\ \alpha_3 : A_3 \sqsubseteq B_1 \sqcap B_2 & \alpha_4 : X \equiv Y \end{array}$$

For $\Sigma = \{A_1, B_1, B_2, X, Y\}$, the best 1-excerpt is $\{\alpha_4\}$ and the best 3-excerpt is $\{\alpha_1, \alpha_2, \alpha_3\}$.

To preserve the largest possible amount of semantic information in a k -excerpt, it would be preferable to extract k -excerpts that have the lowest cDiff -value among all the subsets of size k . However, it is difficult in general to compute all such excerpts in an exhaustive way as all the $\binom{|\mathcal{O}|}{k}$ subsets of size k would have to be enumerated. The following experiment and example show that we need an alternative approach to computing best ontology excerpts instead of enumerating all possible ontology excerpts.

We performed an experiment on the ontology MESH¹, for which we randomly extracted excerpts and computed their cDiff -values w.r.t. a considered signature containing 5000 concept names and

¹<http://biportal.bioontology.org/ontologies/MESH>

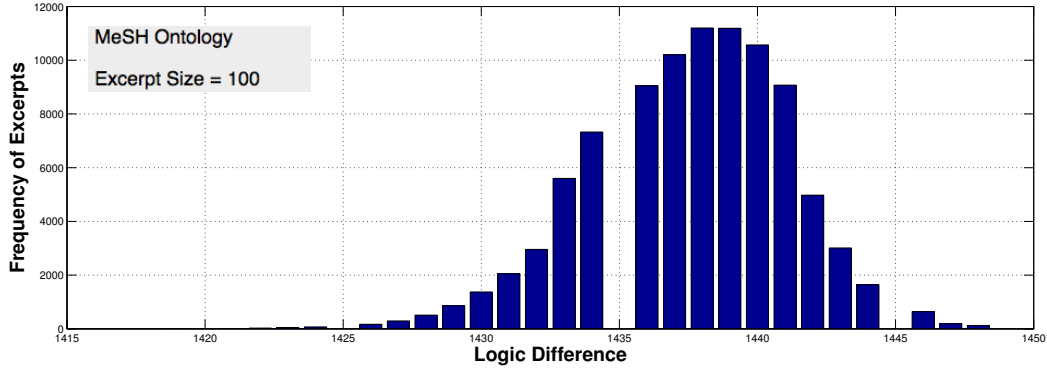


Figure 5.1 Distribution of 93170 Random Excerpts over Respective cDiff-Values

no role names and the corresponding locality-based module contained 1610 axioms. To limit the search space, we selected a subset of MESH containing 2491 axioms, from which we randomly extracted 93 170 many k -excerpts, for $k = 100$. Indeed, for an ontology of that size and a k -value of 100, there exist around 6.2×10^{180} k -excerpts, which renders an exhaustive search through all the excerpts impossible. The results that we obtained are summarized in Table 5.1. The total number of possible k -excerpts is given in the first column, and the cDiff-values that we observed, except for the value 1416, were regrouped into several intervals that are shown in the 6 right-most columns of the table. The percentage of k -excerpts whose cDiff-values fell into the respective intervals is shown in the second row of these columns. The results show that the performance of ontology excerpts varies a lot. It is necessary to develop an algorithm to compute approximate or even best excerpts.

IR Excerpt

We introduced an approximate approach to compute excerpts, called IR excerpts [9], based on the technique of Information Retrieval (IR).

In the field of IR, vector representations of documents and queries are a fundamental tool for modelling problems, based on which different retrieval strategies can be applied. We first define the vector representation for axioms and signatures.

In the remainder of this section, we assume that every ontology \mathcal{O} is associated with a strict total order \prec on the elements of $\text{sig}(\mathcal{O})$. Whenever we want to access the i -th signature element of \mathcal{O} we refer to the i -element w.r.t. the assumed order \prec , starting from the smallest element. For a signature $\Sigma \subseteq \text{sig}(\mathcal{O})$ or axiom $\alpha \in \mathcal{O}$, we can define the signature vector of Σ and the axiom vector of α as follows.

Definition 113 (Signature and Axiom Vector) For a signature $\Sigma \subseteq \text{sig}(\mathcal{O})$, the signature vector of Σ , written $\vec{\Sigma} = [v_1, v_2, \dots]$, is a vector of length $|\text{sig}(\mathcal{O})|$ such that $v_i = 1$ if the i -th element of $\text{sig}(\mathcal{O})$ appears in Σ , otherwise $v_i = 0$. Similarly, for an axiom $\alpha \in \mathcal{O}$ we define $\vec{\alpha} = \overline{\text{sig}(\alpha)}$.

Example 114 (Ex. 111 contd.) Let \mathcal{O} be the ontology defined as in Example 25, and let $\Sigma = \{A, B, r\}$. We assume the strict total order $\prec \subseteq \text{sig}(\mathcal{O}) \times \text{sig}(\mathcal{O})$ given by $A \prec B \prec X \prec Y \prec r \prec s$. Then we obtain the following signature vector for Σ and axiom vectors for each axiom of \mathcal{O} :

$$\begin{aligned} \vec{\Sigma} &= [1, 1, 0, 0, 1, 0] & \vec{\alpha}_1 &= [1, 1, 1, 0, 1, 0] & \vec{\alpha}_2 &= [1, 1, 0, 0, 0, 0] \\ & & \vec{\alpha}_3 &= [1, 1, 1, 0, 0, 0] & \vec{\alpha}_4 &= [1, 1, 1, 1, 1, 1] \end{aligned}$$

Then we can define the distance of an axiom and a set of signature by the distances measures between the axiom and signature vectors. A first measure is the cosine value, resulting in the COS-k-module.

Definition 115 (COS-distance between Axiom and Signature) Given an axiom α and a signature set Σ , the COS-distance between α and Σ is defined as follows:

$$d_{\text{cos}}(\alpha, \Sigma) = \cos(\vec{\alpha}, \vec{\Sigma}) = \frac{\sum_{i=1}^n x_i y_i}{\sqrt{\sum_{i=1}^n x_i^2} \sqrt{\sum_{i=1}^n y_i^2}},$$

where $\vec{\alpha} = [x_1, x_2, \dots, x_n]$ and $\vec{\Sigma} = [y_1, y_2, \dots, y_n]$.

Example 116 (Ex. 114 contd.) Let \mathcal{O} be the ontology defined as in Example 25, let \prec be the total order on $\text{sig}(\mathcal{O})$ as defined in Example 114, and let $\Sigma = \{A, B, r\}$. Then we have that:

$$\begin{aligned} d_{\text{cos}}(\alpha_1, \Sigma) &= 3/(\sqrt{4}\sqrt{3}) \approx 0.8660 & d_{\text{cos}}(\alpha_2, \Sigma) &= 2/(\sqrt{2}\sqrt{3}) \approx 0.8164 \\ d_{\text{cos}}(\alpha_3, \Sigma) &= 2/(\sqrt{3}\sqrt{3}) \approx 0.6667 & d_{\text{cos}}(\alpha_4, \Sigma) &= 3/(\sqrt{6}\sqrt{3}) \approx 0.707 \end{aligned}$$

Therefore, the ranking of the axioms will be $\alpha_1 \triangleright \alpha_2 \triangleright \alpha_4 \triangleright \alpha_3$.

In general, IR-excerpts are computed mainly by the following three steps:

- (i) represent each axiom and signature as a vector;
- (ii) compute the cosine similarity (COS) value between each axiom vector and a signature vector;
- (iii) rank all axioms according to its similarity value and the top- k ranked axioms are IR k -excerpts;

Obviously, the ranking based on vector similarity is stable so that IR-excerpts are monotonic with respect to a given terminology and a signature.

Figure 5.1 shows the distribution of the gain value over the 93 170 excerpts of the MESH-fragment, i.e. each bar in the chart shows the number of excerpts that have the cDiff-value shown on the x -axis that is associated with the bar (no excerpts having an cDiff-value of 1 435 or 1 445 were found). We note that the excerpt extracted using the IR-technique had a Gain value (lowest cDiff-value of 1 416) that was higher than the values of all the random excerpts we extracted.

Judging from the experimental results that we obtained so far, one could draw the conclusion that excerpts produced by the IR-technique appear to result in low cDiff-values in general. To test this

Nr. of Excerpts	cDiff-Value Intervals					
	[1419, 1420]	[1421, 1425]	[1426, 1430]	[1431, 1435]	[1436, 1440]	[1441, 1448]
6.2×10^{180}	2.14×10^{-5}	0.16	3.45	19.26	56.03	21.10

Table 5.1 Percentage of k -Excerpts Falling into Various cDiff-Value Intervals

hypothesis, we conducted another experiment in which we limited the size of the ontology in such a way that an exhaustive enumeration of all its excerpts is feasible.

We performed an exhaustive computation of all the k -excerpts, with $1 \leq k \leq 19$, together with the cDiff-values of a fragment \mathcal{O}_f of SNOMED CT that contains 19 axioms, using $\Sigma = \text{sig}(\mathcal{O}_f)$ as signature. For every $1 \leq k \leq 19$ we also computed the excerpts returned by the IR method for Σ . The results that we obtained are shown in Table 5.2. The first column indicates the value of k and the total number of possible k -excerpts is given in the second column. The 24 cDiff-values that we observed were then regrouped into 8 intervals of three elements, and the percentage of k -excerpts whose cDiff-value fell into the respective intervals is shown in the last 8 columns. The interval that contained the cDiff-value for the excerpt computed by the IR-method is indicated using a background coloured in gray. One can see that in none of the cases for $k < 19$, the k -excerpt obtained using the IR-based technique had the lowest cDiff-value. In other words, the IR-based technique fails to extract the best excerpt for $k < 19$.

The previous experiment has thus established that our hypothesis was wrong, i.e., the IR-based technique cannot guarantee to find the best excerpts in every case. Moreover, we can derive an ever stronger conclusion using the following example.

Example 117 Let \mathcal{O} consist of the following three axioms:

$$\alpha_1 : A_1 \sqsubseteq B_1 \sqcap \exists r.X, \quad \alpha_2 : A_3 \sqsubseteq A_2 \sqcap B_3, \quad \alpha_3 : A_2 \sqsubseteq B_2$$

Let $\Sigma = \text{sig}(\mathcal{O})$. Then the cDiff-values for all 1- and 2-excerpts of \mathcal{O} are respectively shown in the left- and right-hand side of the table below.

	$\{\alpha_1\}$	$\{\alpha_2\}$	$\{\alpha_3\}$		$\{\alpha_1, \alpha_2\}$	$\{\alpha_1, \alpha_3\}$	$\{\alpha_2, \alpha_3\}$
cDiff	4	5	6		3	4	2

The COS-distance between each of the three axioms α_i and Σ is as follows (using an implicit order on the signature elements): $d_{\text{cos}}(\alpha_1, \Sigma) \approx 0.707$, $d_{\text{cos}}(\alpha_2, \Sigma) \approx 0.612$, $d_{\text{cos}}(\alpha_3, \Sigma) = 0.5$. Thus, we obtain the following IR-ranking for the axioms: $\alpha_1 \triangleright \alpha_2 \triangleright \alpha_3$. Although the best 1-excerpt is $\{\alpha_1\}$, the best 2-excerpt is given by $\{\alpha_2, \alpha_3\}$ without having the highest ranked axiom α_1 .

k	Nr. of Excerpts	cDiff-Value Intervals							
		[0, 2]	[3, 5]	[6, 8]	[9, 11]	[12, 14]	[15, 17]	[18, 20]	[21, 23]
1	19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00
2	171	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00
3	969	0.00	0.00	0.00	0.00	0.00	0.00	0.10	99.90
4	3 876	0.00	0.00	0.00	0.00	0.00	0.00	0.52	99.48
5	11 628	0.00	0.00	0.00	0.00	0.00	0.02	1.52	98.46
6	27 132	0.00	0.00	0.00	0.00	0.00	0.10	3.46	96.44
7	50 388	0.00	0.00	0.00	0.00	0.03	0.34	6.67	92.97
8	75 582	0.00	0.00	0.00	0.00	0.10	0.87	11.45	87.58
9	92 378	0.00	0.00	0.00	0.00	0.31	1.94	17.89	79.87
10	92 378	0.00	0.00	0.00	0.00	0.78	3.96	25.58	69.68
11	75 582	0.00	0.00	0.00	0.00	1.78	7.63	33.31	57.28
12	50 388	0.00	0.00	0.00	0.05	3.83	13.79	38.80	43.52
13	27 132	0.00	0.00	0.00	0.35	8.10	22.56	39.18	29.81
14	11 628	0.00	0.00	0.01	1.94	16.10	31.49	32.68	17.78
15	3 876	0.00	0.00	0.70	7.22	27.73	34.73	20.92	8.69
16	969	0.00	0.00	4.75	19.09	37.36	26.73	8.98	3.10
17	171	0.00	2.34	18.71	34.50	31.58	10.53	1.75	0.58
18	19	0.00	36.84	31.58	26.32	5.26	0.00	0.00	0.00
19	1	100.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 5.2 Percentage of k -Excerpts Falling into Various cDiff-Value Intervals

5.2 Encoding Approach of Computing Best Excerpt

In this section, we aim to find an exact algorithm to compute best excerpts. We achieve this based on subsumption justifications introduced in Chapter 3 together with an encoding of the best k -excerpt problem into a partial Max-SAT problem, with the aim of delegating the task of finding the best excerpt to a Max-SAT solver. In that way we can leverage the decades of research efforts dedicated to developing efficient SAT solvers for our problem setting. We continue with reviewing basic notions relating to propositional logic and Max-SAT.

The objective of a partial Max-SAT problem is hence to find a propositional valuation that satisfies all the hard clauses in H and that satisfies a maximal number of the soft clauses in S . Note that a partial Max-SAT problem may nevertheless admit several solutions.

Note that our encoding requires that a cover of the $\langle A, \Sigma \rangle$ -subsumption justifications has already been computed for every concept name $A \in \Sigma$. Moreover, our partial Max-SAT encoding is language agnostic, i.e., it works for any TBox as long as its subsumption justifications can be computed.

As we are interested in finding a subset $\mathcal{M} \subseteq \mathcal{T}$ of a TBox \mathcal{T} that fulfills certain properties, we encode the presence of an axiom $\alpha \in \mathcal{T}$ in \mathcal{M} using a propositional variable p_α . Intuitively, in a solution v to our partial Max-SAT problem we will have that $v(p_\alpha) = 1$ if, and only if, the axiom α is contained in the best k -excerpt.

First, we define the set $\Upsilon_\Sigma^\mathcal{L}$. Let \mathcal{L} be the language that the terminology is formulated in and let Σ be a signature. We set $\Upsilon_\Sigma^\mathcal{L}$ to be the set $\text{Wtn}_\Sigma(\mathcal{T}_1, \mathcal{T}_2)$, where \mathcal{T}_1 and \mathcal{T}_2 are \mathcal{L} -terminologies. Then it follows from the primitive witnesses theorems in [23] that

- $\Upsilon_\Sigma^{\mathcal{L}} = \Sigma \cap \text{N}_C$,
- $\Upsilon_\Sigma^{\mathcal{L}\mathcal{H}} = \Sigma$,
- $\Upsilon_\Sigma^{\mathcal{L}\mathcal{H}^r} = \Sigma \cup \{ \text{dom}(r), \text{ran}(r) \mid r \in \Sigma \cap \text{N}_R \}$.

For encoding the set of all $\langle \sigma, \Sigma \rangle$ -subsumption justifications \mathbb{J}_σ ($\sigma \in \Sigma$), we next introduce an encoding method for a sub-terminology.

Definition 118 (Encoding of Terminology Subsets) *Let \mathcal{T} be an \mathcal{L} -terminology. For every axiom $\alpha \in \mathcal{T}$, let $p_\alpha \in \Pi$ be a fresh propositional variable.*

For a subset $\mathcal{M} \subseteq \mathcal{T}$, we define the propositional formula $F_\mathcal{M}$ that encodes \mathcal{M} as $F_\mathcal{M} := \bigwedge_{\alpha \in \mathcal{M}} p_\alpha$. Moreover, for a set $\mathbb{M} \subseteq 2^\mathcal{T}$, we define the propositional formula $G_\mathbb{M}$ associated with \mathbb{M} as

$$G_\mathbb{M} := \bigvee_{\mathcal{M} \in \mathbb{M}} F_\mathcal{M} = \bigvee_{\mathcal{M} \in \mathbb{M}} \left(\bigwedge_{\alpha \in \mathcal{M}} p_\alpha \right).$$

Example 119 *Let $\mathcal{T} = \{ \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5 \}$, $\mathcal{M} = \{ \alpha_1, \alpha_2, \alpha_3 \}$, and $\mathbb{M} = \{ \{ \alpha_2, \alpha_3 \}, \{ \alpha_1, \alpha_4 \} \}$. Then $F_\mathcal{M} = p_{\alpha_1} \wedge p_{\alpha_2} \wedge p_{\alpha_3}$ and $G_\mathbb{M} = (p_{\alpha_2} \wedge p_{\alpha_3}) \vee (p_{\alpha_1} \wedge p_{\alpha_4})$.*

Definition 120 (Encoding of the Best Excerpt Problem) *For every $\sigma \in \Upsilon_\Sigma^\mathcal{L}$, let \mathbb{J}_σ be the set of all the $\langle \sigma, \Sigma \rangle$ -subsumption justifications of an \mathcal{L} -terminology \mathcal{T} , and let q_σ be a fresh propositional variable. The partial Max-SAT problem for finding best k -excerpts of \mathcal{T} w.r.t. Σ , denoted with $P_k(\mathcal{T}, \Sigma)$, is defined as follows. We set $P_k(\mathcal{T}, \Sigma) := (H_k(\mathcal{T}), S_k(\mathcal{T}, \Sigma))$, where*

$$H_k(\mathcal{T}) := \text{Card}(\mathcal{T}, k) \cup \bigcup_{\sigma \in \Upsilon_\Sigma^\mathcal{L}} \text{Clauses}(q_\sigma \leftrightarrow G_{\mathbb{J}_\sigma}),$$

$$S_k(\mathcal{T}, \Sigma) := \{ q_\sigma \mid \sigma \in \Upsilon_\Sigma^\mathcal{L} \},$$

and $\text{Card}(\mathcal{T}, k)$ is the set of clauses specifying that at most k clauses from the set $\{ p_\alpha \mid \alpha \in \mathcal{T} \}$ must be satisfied.

In the hard part of our partial Max-SAT problem, the clauses in $\text{Card}(\mathcal{T}, k)$ specify that the cardinality of the resulting excerpt $\mathcal{E} \subseteq \mathcal{T}$ must be equal to k . We do not fix a certain encoding that should be used to obtain $\text{Card}(\mathcal{T}, k)$, but we note that there exist several techniques that require a

polynomial number of clauses in k and in the size of \mathcal{T} (see e.g. [44]). Moreover, for every sigma symbol $\sigma \in \Upsilon_{\Sigma}^{\mathcal{L}}$, the variable q_{σ} is set to be equivalent to the formula $G_{\mathbb{J}}$, i.e. q_{σ} will be satisfied in a valuation if and only if the resulting excerpt will have the property that the knowledge of σ w.r.t. Σ in \mathcal{T} is preserved. Note that $\sigma \in \Upsilon_{\Sigma}^{\mathcal{L}}$, which depends on the description language that \mathcal{T} is formulated on. Finally, the set $S_k(\mathcal{T}, \Sigma)$ of soft clauses specifies that a maximal number of q_{σ} must be satisfied, enforcing that the resulting excerpt \mathcal{E} will yield the smallest possible number of difference witnesses (whilst obeying the constraint that $|\mathcal{E}| \leq k$).

Example 121 *Let \mathcal{T} consist of the following four axioms:*

$$\begin{array}{ll} \alpha_1 := A \sqsubseteq X_1 \sqcap X_2 \sqcap Y & \alpha_2 := X_1 \sqsubseteq B \\ \alpha_3 := X_2 \sqsubseteq B & \alpha_4 := r \sqsubseteq s \end{array}$$

Let $\Sigma = \{A, B, Y, r, s\}$ be a signature. Then, for $\sigma \in \Upsilon_{\Sigma}^{\mathcal{L}\mathcal{H}} = \Sigma$, we obtain the following $\langle \sigma, \Sigma \rangle$ -subsumption justifications \mathbb{J}_{σ} :

$$\mathbb{J}_A = \{\{\alpha_1, \alpha_2\}, \{\alpha_1, \alpha_3\}\},$$

$$\mathbb{J}_B = \{\{\alpha_1, \alpha_2\}, \{\alpha_1, \alpha_3\}\},$$

$$\mathbb{J}_Y = \{\{\alpha_1\}\},$$

$$\mathbb{J}_r = \{\{\alpha_4\}\},$$

$$\mathbb{J}_s = \{\{\alpha_4\}\}.$$

Each subsumption justification \mathbb{J}_{σ} is encoded as a propositional logic formula \mathcal{G}_{σ} as follows:

$$\mathcal{G}_A = ((p_1 \wedge p_2) \vee (p_1 \wedge p_3)),$$

$$\mathcal{G}_B = ((p_1 \wedge p_2) \vee (p_1 \wedge p_3)),$$

$$\mathcal{G}_Y = p_1,$$

$$\mathcal{G}_r = p_4,$$

$$\mathcal{G}_s = p_4.$$

By using SAT solver, we obtain the assignments of truth values for the propositional variables in the tuple $\langle p_1, p_2, p_3, p_4 \rangle$ (in this order) for different k value:

- when $k=1$: $\langle 0, 0, 0, 1 \rangle$;
- when $k=2$: $\langle 1, 0, 0, 1 \rangle$;
- when $k=3$: $\langle 1, 1, 0, 1 \rangle$ and $\langle 1, 0, 1, 1 \rangle$;

- when $k=4$: $\langle 1, 1, 1, 1 \rangle$.

Finally, we have that for Σ , the best 1-excerpt is $\{\alpha_4\}$, the best 2-excerpt is $\{\alpha_1, \alpha_4\}$, the best 3-excerpts are $\{\alpha_1, \alpha_2, \alpha_4\}$ and $\{\alpha_1, \alpha_3, \alpha_4\}$, and the best 4-excerpt is \mathcal{T} .

The encoding has the following properties.

Lemma 122 *Let \mathcal{T} be an \mathcal{L} -terminology and Σ be a signature. Additionally, let $0 \leq k \leq |\mathcal{T}|$ and let v be a valuation that satisfies $H_k(\mathcal{T})$.*

Then it holds that for $\mathcal{E} := \{\alpha \in \mathcal{T} \mid v(p_\alpha) = 1\}$, the following statement holds:

$$\text{cDiff}_\Sigma(\mathcal{T}, \mathcal{E}) = \{\sigma \in \Upsilon_\Sigma^\mathcal{L} \mid v(q_\sigma) = 0\}.$$

Proof It is not difficult to get the conclusions that the following statements are equivalent:

1. $\sigma \in \text{cDiff}_\Sigma(\mathcal{T}, \mathcal{E})$.
2. For every $\mathcal{J} \in \mathbb{J}_\sigma$, $\mathcal{J} \notin \mathcal{E}$.
3. For each \mathcal{J} , there exists an axiom $\alpha \in \mathcal{T} \cap \mathcal{J}$ such that $v(\alpha) = 0$. i.e., $v(\bigwedge_{\alpha \in \mathcal{J}} p_\alpha) = 0$.
4. $v(G_{\mathbb{J}_\sigma}) = v(\bigvee_{\mathcal{J} \in \mathbb{J}_\sigma} (\bigwedge_{\alpha \in \mathcal{J}} p_\alpha)) = 0$.
5. $v(q_\sigma) = 0$.
6. $\sigma \in \{\sigma \in \Upsilon_\Sigma^\mathcal{L} \mid v(q_\sigma) = 0\}$.

The equivalence between the statements 4 and 5 is based on the fact that $\text{Clauses}(q_\sigma \leftrightarrow G_{\mathbb{J}_\sigma})$ is in the hard part of the encoding. □

Lemma 123 *Let \mathcal{T} be an \mathcal{L} -terminology and let Σ be a signature. Additionally, let $0 \leq k \leq |\mathcal{T}|$ and let $\mathcal{E} \subseteq \mathcal{T}$ with $|\mathcal{E}| = k$. Then there exists a valuation v such that $\mathcal{E} = \{\alpha \in \mathcal{T} \mid v(p_\alpha) = 1\}$, v satisfies $H_k(\mathcal{T})$ and $\Upsilon_\Sigma^\mathcal{L} \setminus \text{cDiff}_\Sigma(\mathcal{T}, \mathcal{E}) = \{\sigma \in \Upsilon_\Sigma^\mathcal{L} \mid v(q_\sigma) = 1\}$.*

Proof Let v be a valuation defined by setting: first, for every $\alpha \in \mathcal{T}$, let $v(p_\alpha) = 1$ iff $\alpha \in \mathcal{E}$. Additionally, for every $\sigma \in \Upsilon_\Sigma^\mathcal{L}$ we set $v(q_\sigma) = 1$ iff $v(G_{\mathbb{J}_\sigma}) = 1$. Finally, We assume that v is extended to the additional variables in $\text{Card}(\mathcal{T}, k)$ and $\bigcup_{\sigma \in \Upsilon_\Sigma^\mathcal{L}} \text{Clauses}(q_\sigma \leftrightarrow G_{\mathbb{J}_\sigma})$ such that v satisfies $\text{Card}(\mathcal{T}, k)$ and $\bigcup_{\sigma \in \Upsilon_\Sigma^\mathcal{L}} \text{Clauses}(q_\sigma \leftrightarrow G_{\mathbb{J}_\sigma})$. Note that such an extension is possible as exactly k variables from $\{p_\alpha \mid \alpha \in \mathcal{T}\}$ are initially set to 1.

Now it follows from Lemma 122 that $\Upsilon_\Sigma^\mathcal{L} \setminus \text{cDiff}_\Sigma(\mathcal{T}, \mathcal{E}) = \{\sigma \in \Upsilon_\Sigma^\mathcal{L} \mid v(q_\sigma) = 1\}$. □

We can now show the correctness of our encoding, i.e. a best k -excerpt can be obtained from a solution to the partial Max-SAT problem $P_k(\mathcal{T}, \Sigma)$.

Theorem 124 (Correctness & Completeness) *Let \mathcal{T} be an \mathcal{L} -terminology, let Σ be a signature, and let $0 \leq k \leq |\mathcal{T}|$. It holds that $\mathcal{E} \subseteq \mathcal{T}$ is a best k -excerpt of \mathcal{T} w.r.t. Σ iff there exists a solution v of the partial Max-SAT problem $P_k(\mathcal{T}, \Sigma)$ such that $\mathcal{E} = \{\alpha \in \mathcal{T} \mid v(p_\alpha) = 1\}$.*

Proof First we prove the correctness.

Let $\mathcal{M} := \{ \alpha \in \mathcal{T} \mid v(p_\alpha) = 1 \}$. We have to show that there does not exist a set $\mathcal{E} \subseteq \mathcal{T}$ with $|\mathcal{E}| \leq k$ and $|\text{cDiff}_\Sigma(\mathcal{T}, \mathcal{E})| < |\text{cDiff}_\Sigma(\mathcal{T}, \mathcal{M})|$. We now assume towards a contradiction that there exists a set $\mathcal{E} \subseteq \mathcal{T}$ with $|\mathcal{E}| \leq k$ and $|\text{cDiff}_\Sigma(\mathcal{T}, \mathcal{E})| < |\text{cDiff}_\Sigma(\mathcal{T}, \mathcal{M})|$.

Note that $v(q_\sigma) = 1$ implies $\sigma \notin \text{cDiff}_\Sigma(\mathcal{T}, \mathcal{M})$ for every $\sigma \in \Upsilon_\Sigma^\mathcal{L}$.

Then by Lemma 123 there exists a valuation v' such that $\mathcal{E} = \{ \alpha \in \mathcal{T} \mid v'(p_\alpha) = 1 \}$, v' satisfies $H_k(\mathcal{T})$ and $\Upsilon_\Sigma^\mathcal{L} \setminus \text{cDiff}_\Sigma(\mathcal{T}, \mathcal{E}) = \{ A \in \Sigma \cap \text{Nc} \mid v'(q_A) = 1 \}$.

From $|\text{cDiff}_\Sigma(\mathcal{T}, \mathcal{E})| < |\text{cDiff}_\Sigma(\mathcal{T}, \mathcal{M})|$, we can infer that

$$|\Upsilon_\Sigma^\mathcal{L}| - |\text{cDiff}_\Sigma(\mathcal{T}, \mathcal{M})| < |\Upsilon_\Sigma^\mathcal{L}| - |\text{cDiff}_\Sigma(\mathcal{T}, \mathcal{E})|.$$

Hence, by Lemma 122 we have

$$|\{ \sigma \in \Upsilon_\Sigma^\mathcal{L} \mid v(q_\sigma) = 1 \}| < |\{ \sigma \in \Upsilon_\Sigma^\mathcal{L} \mid v'(q_\sigma) = 1 \}|,$$

which contradicts v being a solution of the partial Max-Sat problem $P_k(\mathcal{T}, \Sigma)$ (Item (ii) of Definition 49).

Then we prove the completeness. Lemma 123 shows that there exists a valuation v such that $\mathcal{E} = \{ \alpha \in \mathcal{T} \mid v(p_\alpha) = 1 \}$, v satisfies $H_k(\mathcal{T})$ and $\Upsilon_\Sigma^\mathcal{L} \setminus \text{cDiff}_\Sigma(\mathcal{T}, \mathcal{E}) = \{ \sigma \in \Upsilon_\Sigma^\mathcal{L} \mid v(q_\sigma) = 1 \}$.

We assume towards a contradiction that there exists a valuation v' that satisfies $H_k(\mathcal{T})$ and

$$|\{ A \in \Upsilon_\Sigma^\mathcal{L} \mid v(q_A) = 1 \}| < |\{ A \in \Upsilon_\Sigma^\mathcal{L} \mid v'(q_A) = 1 \}|.$$

Let $\mathcal{E}' := \{ \alpha \in \mathcal{T} \mid v'(p_\alpha) = 1 \}$. Hence, by Lemma 122 it holds that

$$|\Upsilon_\Sigma^\mathcal{L}| - |\text{cDiff}_\Sigma(\mathcal{T}, \mathcal{E}')| < |\Upsilon_\Sigma^\mathcal{L}| - |\text{cDiff}_\Sigma(\mathcal{T}, \mathcal{E})|.$$

Finally, it would hold that $|\text{cDiff}_\Sigma(\mathcal{T}, \mathcal{E}')| < |\text{cDiff}_\Sigma(\mathcal{T}, \mathcal{E})|$, contradicting the fact that \mathcal{E} is a best k -excerpt of \mathcal{T} w.r.t. Σ (Definition 110). \square

Algorithm 16 shows how best excerpts are computed by using partial Max-SAT encoding. In Line 9, the algorithm iterates over every $\sigma \in \Upsilon_\Sigma^\mathcal{L}$ and the set of all subsumption justifications \mathbb{J}_σ are computed. The formula $G_{\mathbb{J}_\sigma}$ is computed next and stored in a set S . After the iteration over all element σ in $\Upsilon_\Sigma^\mathcal{L}$ is complete, the partial Max-SAT problem $P_k(\mathcal{T}, \Sigma)$ is constructed with the help of the formulas $G_{\mathbb{J}_\sigma}$ that are stored in S . Subsequently, a solution v of $P_k(\mathcal{T}, \Sigma)$ is computed using a partial Max-SAT solver and the best k -excerpt is returned by analysing which variables p_α have been set to 1 in the valuation v .

Algorithm 16: Computing Best k -Excerpts for an \mathcal{L} -terminology

```

1 function ComputeBestExcerpt ( $\mathcal{T}, \Sigma, k$ )
2   if  $k = 0$  then
3     return  $\emptyset$ 
4   end
5   if  $k = |\mathcal{T}|$  then
6     return  $\mathcal{T}$ 
7   end
8    $S := \emptyset$ 
9   for every  $\sigma \in \Upsilon_{\Sigma}^{\mathcal{L}}$  do
10    Compute  $\langle \sigma, \Sigma \rangle$ -subsumption justifications of  $\mathcal{T}$ :  $\mathbb{J}_{\sigma}$ 
11    Transfer  $\mathbb{J}_{\sigma}$  to its propositional formula  $G_{\mathbb{J}_{\sigma}}$ 
12     $S := S \cup \{G_{\mathbb{J}_{\sigma}}\}$ 
13  end
14  Compute  $P_k(\mathcal{T}, \Sigma)$  using  $S$ 
15  Find the set of solutions  $V$  of  $P_k(\mathcal{T}, \Sigma)$  using partial Max-SAT solver
16  return  $\{\{\alpha \in \mathcal{T} \mid v(p_{\alpha}) = 1, v \in V\}\}$ 

```

As Proposition 92 shows, the computation of all subsumption justifications runs in EXPTIME in the size of \mathcal{T} . Note that the cardinality encoding $\text{Card}(\mathcal{T}, k)$ can be kept in linear size of \mathcal{T} and k [44]. However, the second part of the hard clauses $\text{Clauses}(q_{\sigma} \leftrightarrow G_{\mathbb{J}_{\sigma}})$ can be exponential due to the fact that the number of subsumption justifications of each σ is exponential in the worst case. Therefore, we can construct a non-deterministic algorithm to solve the encoding problem in EXPTIME, that is, solving the encoding problem is in NEXPTIME. Together with the EXPTIME-upper bound for pre-computing all subsumption justifications, the whole problem of computing best excerpts is NEXPTIME. We arrive at the following conclusion.

Proposition 125 *Let \mathcal{T} be an \mathcal{L} -terminology, let Σ be a signature, and let $0 \leq k \leq |\mathcal{T}|$. Then checking whether a subset $\mathcal{E} \subseteq \mathcal{T}$ is a best k -excerpt of \mathcal{T} w.r.t. Σ is in NEXPTIME.*

5.3 Computing Best Excerpt for Weighted Signatures

Best k -excerpts provide users with a way to extract the most relevant axioms from a big ontology with respect to a given signature. However, in certain cases, concepts or roles in the signature can have different importance. How to reflect such importance differences among signature elements, in this section, we extend the best excerpt notion to the case of weighted signatures. For this, we propose a new incompleteness measure considering a weighted signature and study the corresponding best k -excerpts.

First, we introduce the definition of weighted signature.

Definition 126 (Weighted Signature) A weight function is a function $w : \mathbb{N}_C \cup \mathbb{N}_R \rightarrow \mathbb{R} \cup \{\infty\}$ mapping a symbol $\sigma \in \mathbb{N}_C \cup \mathbb{N}_R$ to a value $w(\sigma) \in \mathbb{R}$ or to $w(\sigma) = \infty$. Given a signature Σ and a weight function w , a weighted signature Σ_w is the set $\Sigma_w = \{(\sigma, w(\sigma)) \mid \sigma \in \Sigma\}$.

A signature Σ is *weighted uniformly under a weight function w* if, and only if, w assigns the same value to every symbol in Σ , i.e. $w(\sigma_1) = w(\sigma_2)$ for every $\sigma_1, \sigma_2 \in \Sigma$. For a weight function w whose range equals $\{1.0\}$ (i.e., w maps every symbol to 1.0), we also write Σ° instead of Σ_w . For instance, $\Sigma_w = \{(A, 2.0), (B, 0.9), (r, 1.3)\}$ is the weighted signature for the signature $\Sigma = \{A, B, r\}$ and a weight function w satisfying $w(A) = 2.0$, $w(B) = 0.9$ and $w(r) = 1.3$. On the other hand, a weight function w' satisfying $w'(A) = w'(B) = w'(r) = 1.4$ weighs the symbols in Σ uniformly resulting in $\Sigma_{w'} = \{(A, 1.4), (B, 1.4), (r, 1.4)\}$, whereas $\Sigma^\circ = \{(A, 1.0), (B, 1.0), (r, 1.0)\}$.

For best k -excerpts under a weighted signature Σ_w , we use, as an incompleteness measure $\mu_{\mathcal{T}}^{\Sigma_w}$ for Σ_w and \mathcal{T} , the sum of the weights of the $\mathcal{E}\mathcal{L}$ -concept inclusion difference witnesses and role inclusion difference witnesses in Σ w.r.t. \mathcal{T} and \mathcal{E} , formally:

$$\mu_{\mathcal{T}}^{\Sigma_w}(\mathcal{E}) = \sum_{\sigma \in \text{Wtn}_{\Sigma}(\mathcal{T}, \mathcal{E})} w(\sigma).$$

For computing the best k -excerpts for a weighted signature, similar to the case of Σ° (equivalent to unweighted case), we present an encoding approach, but to a *weighted* partial Max-SAT problem, with the aim of delegating the task of finding the best excerpt to a Max-SAT solver.

Definition 127 (Encoding of the Best Excerpt Problem for Weighted Signature) Let \mathcal{T} be an \mathcal{L} -terminology, let Σ_w be a weighted signature, and let $0 \leq k \leq |\mathcal{T}|$. Additionally, for every $\sigma \in \Upsilon_{\Sigma_w}^{\mathcal{L}}$, let \mathbb{J}_σ be the set of the $\langle \sigma, \Sigma_w \rangle$ -subsumption justifications of \mathcal{T} , and let q_σ be a fresh propositional variable.

The weighted partial Max-SAT problem for finding best k -excerpts of \mathcal{T} w.r.t. Σ_w , denoted with $P_k(\mathcal{T}, \Sigma_w)$, is defined as follows.

We set $P_k(\mathcal{T}, \Sigma_w) := (H_k(\mathcal{T}), S_k(\mathcal{T}, \Sigma_w))$, where

$$H_k(\mathcal{T}) := \text{Card}(\mathcal{T}, k) \cup \bigcup_{\sigma \in \Upsilon_{\Sigma_w}^{\mathcal{L}}} \text{Clauses}(q_\sigma \leftrightarrow G_{\mathbb{J}_\sigma}),$$

$$S_k(\mathcal{T}, \Sigma_w) := \{(q_\sigma, w(\sigma)) \mid \sigma \in \Upsilon_{\Sigma_w}^{\mathcal{L}}\},$$

and $\text{Card}(\mathcal{T}, k)$ is a set of clauses specifying that at most k clauses from the set $\{p_\alpha \mid \alpha \in \mathcal{T}\}$ must be satisfied.

In the hard part of our weighted partial Max-SAT problem, the clauses are the same as the unweighted case given in Definition 120.

In the soft part of the problem, the set $S_k(\mathcal{T}, \Sigma_w)$ specifies that the sum of the weights of the concept names in the signature must be maximal by a solution of the problem, enforcing that the

resulting excerpt \mathcal{E} will yield the smallest possible sum of weights of difference witnesses (whilst obeying the constraint that $|\mathcal{E}| \leq k$).

We can now show the correctness of our encoding, i.e. a best k -excerpt for a weighted signature can be exactly obtained from any solution to the weighted partial Max-SAT problem $P_k(\mathcal{T}, \Sigma_w)$.

Theorem 128 (Correctness & Completeness) *Let \mathcal{T} be an \mathcal{L} -terminology, and let Σ_w be a weighted signature. Then: v is a solution of the weighted partial Max-SAT problem $P_k(\mathcal{T}, \Sigma_w)$ iff $\{\alpha \in \mathcal{T} \mid v(p_\alpha) = 1\}$ is a best k -excerpt of \mathcal{T} w.r.t. Σ_w .*

The proof of Theorem 128 is similar to Theorem 124.

Our algorithm for finding a best k -excerpt given a TBox \mathcal{T} and a weighted signature Σ_w (and k with $0 \leq k \leq |\mathcal{T}|$) as input is shown in Algorithm 17. First, the algorithm checks whether $k = 0$ or $k = |\mathcal{T}|$, in which case the empty set or \mathcal{T} itself is the best k -excerpt, respectively. Otherwise, the algorithm continues by computing subsumption justifications for each concept name or role name σ in Σ_w and then transfers them to propositional formula $G_{\mathbb{J}_\sigma}$. After the iteration over all the concept names and role names σ in Σ_w is complete, the weighted partial Max-SAT problem $P_k(\mathcal{T}, \Sigma_w)$ is constructed with the help of the formulas $G_{\mathbb{J}_\sigma}$ that are stored in S . Subsequently, a solution v of $P_k(\mathcal{T}, \Sigma_w)$ is computed using a weighted partial Max-SAT solver and the best k -excerpt is returned by analyzing which variables p_α have been set to 1 in the valuation v .

Similar to Proposition 125, we have the same complexity for computing best excerpts regarding a weighted signature, as stated below.

Proposition 129 *Let \mathcal{T} be an \mathcal{L} -terminology, let Σ_w be a weighted signature, and let $0 \leq k \leq |\mathcal{T}|$. Then checking whether a subset $\mathcal{E} \subseteq \mathcal{T}$ is a best k -excerpt of \mathcal{T} w.r.t. Σ_w is in NEXPTIME.*

Algorithm 17: Computing Best k -Excerpts of \mathcal{T} for Weighted Signature Σ

```

1 function ComputeBestExcerpt ( $\mathcal{T}, \Sigma_w, k$ )
2   if  $k = 0$  then
3     return  $\emptyset$ 
4   end
5   if  $k = |\mathcal{T}|$  then
6     return  $\mathcal{T}$ 
7   end
8    $S := \emptyset$ 
9   for every  $\sigma \in \Upsilon_{\Sigma_w}^{\mathcal{L}}$  do
10    Compute  $\langle \sigma, \Sigma_w \rangle$ -subsumption justifications of  $\mathcal{T}$ :  $\mathcal{J}_{\mathcal{T}}(\sigma, \Sigma_w)$ 
11    Transfer  $\mathcal{J}_{\mathcal{T}}(\sigma, \Sigma_w)$  to its propositional formula  $G_{\mathbb{J}_\sigma}$ 
12     $S := S \cup \{G_{\mathbb{J}_\sigma}\}$ 
13  end
14  Compute  $P_k(\mathcal{T}, \Sigma_w)$  using  $S$ 
15  Find the set of solutions  $V$  of  $P_k(\mathcal{T}, \Sigma_w)$  using weighted partial Max-SAT solver
16  return  $\{\{\alpha \in \mathcal{T} \mid v(p_\alpha) = 1, v \in V\}\}$ 

```

Example 130 Let \mathcal{T} consist of the following five axioms:

$$\begin{aligned} \alpha_1 &:= A \sqsubseteq X_1 \sqcap X_2 \sqcap X_3 \sqcap X_4 & \alpha_2 &:= X_1 \sqsubseteq B_1 \\ \alpha_3 &:= X_2 \sqsubseteq B_1 & \alpha_4 &:= X_3 \sqsubseteq B_2 \\ \alpha_5 &:= X_4 \sqsubseteq B_2 \end{aligned}$$

Let $\Sigma = \{A, B_1, B_2\}$ be a signature. Then, for $\sigma \in \Sigma$, we obtain the following $\langle \sigma, \Sigma \rangle$ -subsumption justifications \mathbb{J}_σ :

$$\mathbb{J}_A = \{\{\alpha_1, \alpha_2\}, \{\alpha_1, \alpha_3\}, \{\alpha_1, \alpha_4\}, \{\alpha_1, \alpha_5\}\},$$

$$\mathbb{J}_{B_1} = \{\{\alpha_1, \alpha_2\}, \{\alpha_1, \alpha_3\}\}, \text{ and}$$

$$\mathbb{J}_{B_2} = \{\{\alpha_1, \alpha_4\}, \{\alpha_1, \alpha_5\}\}.$$

Each subsumption justification \mathbb{J}_σ is encoded as a propositional logic formula \mathcal{G}_σ as follows:

$$\mathcal{G}_A = ((p_1 \wedge p_2) \vee (p_1 \wedge p_3) \vee (p_1 \wedge p_4) \vee (p_1 \wedge p_5)),$$

$$\mathcal{G}_{B_1} = ((p_1 \wedge p_2) \vee (p_1 \wedge p_3)), \text{ and}$$

$$\mathcal{G}_{B_2} = ((p_1 \wedge p_4) \vee (p_1 \wedge p_5)).$$

Now let w_1, w_2, w_3 be weight functions such that

$$w_1(A) = w_1(B_1) = w_1(B_2) = 1.0,$$

$$w_2(A) = w_2(B_1) = 1.0 \text{ and } w_2(B_2) = 2.0, \text{ and}$$

$$w_3(A) = w_3(B_2) = 1.0, w_3(B_1) = 2.0.$$

Feeding the resulting weighted signatures Σ_{w_1} , Σ_{w_2} and Σ_{w_3} together with $k = 2$ to a SAT solver yields the following solutions. For Σ_{w_1} , we obtain four assignments of truth values for the propositional variables in the tuple $\langle p_1, p_2, p_3, p_4, p_5 \rangle$ (in this order): $\langle 1, 1, 0, 0, 0 \rangle$, $\langle 1, 0, 1, 0, 0 \rangle$, $\langle 1, 0, 0, 1, 0 \rangle$ and $\langle 1, 0, 0, 0, 1 \rangle$. Then we decode the assignments to sets of axioms. We obtain the following best 2-excerpts for Σ_{w_1} : $\{\alpha_1, \alpha_2\}$, $\{\alpha_1, \alpha_3\}$, $\{\alpha_1, \alpha_4\}$ and $\{\alpha_1, \alpha_5\}$.

In the case of Σ_{w_2} , however, the solutions returned by the SAT-solver are the assignments $\langle 1, 0, 0, 1, 0 \rangle$ and $\langle 1, 0, 0, 0, 1 \rangle$. Consequently, the best 2-excerpts for Σ_{w_2} are $\{\alpha_1, \alpha_4\}$ and $\{\alpha_1, \alpha_5\}$.

Finally, for Σ_{w_3} , the SAT-solver yields $\langle 1, 1, 0, 0, 0 \rangle$ and $\langle 1, 0, 0, 1, 0 \rangle$ resulting in $\{\alpha_1, \alpha_2\}$ and $\{\alpha_1, \alpha_4\}$ as the best 2-excerpts for Σ_{w_3} .

Example 130 shows that, different weights of Σ -symbols can influence the final results of best excerpts. By giving higher weights on preferring Σ -concept names (like Σ_{w_2} or Σ_{w_3}), users can reduce the number of best k-excerpts. However, in some extreme cases, as shown in the following example where a signature has a much higher weight than the others, the best excerpts for the weighted signature can be totally irrelevant to those w.r.t. Σ° .

Example 131 Let \mathcal{T} consist of the following five axioms:

$$\begin{aligned}\alpha_1 &:= \alpha_1 := A_1 \sqsubseteq X & \alpha_2 &:= X \sqsubseteq Y \\ \alpha_3 &:= Y \sqsubseteq Z & \alpha_4 &:= A_2 \sqsubseteq B_1 \\ \alpha_5 &:= A_3 \sqsubseteq B_2\end{aligned}$$

Let $\Sigma = \{A_1, A_2, A_3, B_1, B_2, Y, Z\}$ be a signature. Now let w_1, w_2 be weight functions such that $w_1(A_1) = w_1(A_2) = w_1(A_3) = w_1(B_1) = w_1(B_2) = w_1(Y) = w_1(Z) = 1.0$ (i.e., Σ_{w_1} is Σ°) and $w_2(A_1) = w_2(A_2) = w_2(A_3) = w_2(Y) = 1.0$ and $w_2(Z) = 9.0$. We obtain that the best 3-excerpt $\mathcal{E}_{\Sigma^\circ}$ for Σ° is $\{\alpha_4, \alpha_5\}$. We have that the best 3-excerpt $\mathcal{E}_{\Sigma_{w_1}}$ for Σ_{w_1} only contains two axioms α_4 and α_5 , as for any best 3-excerpt \mathcal{E}' , $\mu_{\mathcal{T}}^{\Sigma^\circ}(\mathcal{E}') = \mu_{\mathcal{T}}^{\Sigma^\circ}(\{\alpha_4, \alpha_5\})$. The best 3-excerpt $\mathcal{E}_{\Sigma_{w_2}}$ for Σ_{w_2} is $\{\alpha_1, \alpha_2\}$. However, $|\text{Wtn}_{\Sigma^\circ}(\mathcal{T}, \mathcal{E}_{\Sigma^\circ})| = |\{A_1, Y, Z\}| = 3$ and $|\text{Wtn}_{\Sigma_{w_2}}(\mathcal{T}, \mathcal{E}_{\Sigma_{w_2}})| = |\{A_2, A_3, B_1, B_2\}| = 4$.

In Example 131 the value of $w_2(Z)$ is significantly greater than the value of w_2 for the other concept names in the signature. Therefore, $\mathcal{E}_{\Sigma^\circ}$ is not the same as $\mathcal{E}_{\Sigma_{w_2}}$. In such situations, where the weight of a signature symbol is much larger than for the other symbols, the number of difference witnesses has diminishing impact. We aim to remedy the situation in the following proposition, i.e., Proposition 132, by increasing the impact of the number of difference witnesses. This is achieved by ensuring that the weight of any Σ -symbol is smaller than the sum of any other two Σ -symbols.

Proposition 132 Let \mathcal{T} be an \mathcal{L} -terminology, Σ_w be a weighted signature and let Σ° be a uniform signature under the weight function w . Then the following statements hold:

- (i) If $1 \leq w(\sigma) \leq 1 + \frac{1}{|\Upsilon_{\Sigma_w}^{\mathcal{L}}|}$ for every $\sigma \in \Upsilon_{\Sigma_w}^{\mathcal{L}}$, then every best k -excerpt of \mathcal{T} w.r.t. Σ_w will be a best k -excerpt of \mathcal{T} w.r.t. Σ° , but not vice versa.
- (ii) If $\sum_{\sigma_i \in \Sigma_1} w_{\sigma_i} > \sum_{\sigma_j \in \Sigma_2} w_{\sigma_j}$ for any $\Sigma_1, \Sigma_2 \in 2^{\Upsilon_{\Sigma_w}^{\mathcal{L}}}$ and $|\Sigma_1| > |\Sigma_2|$, then every best k -excerpt of \mathcal{T} w.r.t. Σ_w is a best k -excerpt of \mathcal{T} w.r.t. Σ° .

5.4 Ranking Axioms in Best Excerpts

Real-world ontologies are usually too large for human user to browse and comprehend. Best excerpts can help human users to better understand ontologies in order to, say, choose between different ontologies. Ontology excerpts usually consist of more than one axiom. In order to present ontology excerpts in a way that is more friendly to human users, in this section, we present an approach to ranking axioms in excerpts according to a weighted signature.

To obtain an axiom ranking, we first define a notion for weighing axioms.

Definition 133 (Weighted Axiom) An axiom weight function ϕ is a function $\phi : \mathcal{T} \rightarrow \mathbb{R}^+$ mapping an axiom $\alpha \in \mathcal{T}$ to a value $\phi(\alpha) \in \mathbb{R}^+$.

Next we introduce our definition for an axiom ranking. The idea is that an axiom with a larger weight should be ranked at a higher position in the ranking order.

Definition 134 (Axiom Ranking) *Let \mathcal{T} be an \mathcal{L} -terminology, ϕ an axiom weight function and $\alpha, \beta \in \mathcal{T}$ be two axioms. We say that α is preferred over β , denoted as $\alpha \succeq \beta$, iff $\phi(\alpha) \geq \phi(\beta)$.*

Now we present an axiom weight function. Intuitively, an axiom that appears in subsumption justifications of several Σ -symbols with higher weights should receive a higher weight.

Definition 135 *Let \mathcal{T} be a terminology, Σ_w be a signature with a weight function w . Additionally, let \mathbb{J}_σ be the set of the $\langle \sigma, \Sigma_w \rangle$ -subsumption justifications of \mathcal{T} and $J_\sigma = \bigcup_{\mathcal{J} \in \mathbb{J}_\sigma} \mathcal{J}$. The axiom weight function $\phi(\cdot)$ induced by w is defined as following:*

$$\phi_{\Sigma_w}(\alpha) = \sum_{\{\sigma \in \Sigma_w \cap \text{sig}(\alpha) \mid \alpha \in J_\sigma\}} w(\sigma).$$

We denote with \succeq_{Σ_w} the axiom ranking induced by ϕ_{Σ_w} .

Example 136 (Ex. 130 contd.) *Note that α_1 (resp. $\alpha_2, \alpha_3, \alpha_4, \alpha_5$) is contained in at least one of the subsumption justifications of Σ -symbols $\{A, B_1, B_2\}$ (resp. $\{A, B_1\}, \{A, B_1\}, \{A, B_2\}, \{A, B_2\}$). For w_3 , as $\alpha_1 \in J_A, J_{B_1}$ and J_{B_2} , so $\phi(\alpha_1) = w(A) + w(B_1) + w(B_2) = 4.0$. Similarly, we have that $\phi(\alpha_2) = \phi(\alpha_3) = w(A) + w(B_1) = 2.0$, and $\phi(\alpha_4) = \phi(\alpha_5) = w(A) + w(B_2) = 3.0$. Hence, the 2-best excerpts w.r.t. w_3 are $\{\alpha_1, \alpha_2\}$ and $\{\alpha_1, \alpha_4\}$, we can order the axioms by $\alpha_1 \succeq_{\Sigma_w} \alpha_2$ and $\alpha_1 \succeq_{\Sigma_w} \alpha_4$.*

It can readily be seen that the axiom ranking function $\phi(\cdot)$ in Definition 135 gives a total ordering over the axioms in \mathcal{T} w.r.t. to a given weighted signature Σ_w . Note that the axiom weight function only depends on the terminology and weighted signature. For a terminology and a weighted signature, the relative order of two axioms is always the same as long as both axioms are contained in the excerpts, which is easy to see by Definition 135. That is, \succeq is a stable ordering of the axioms given a weighted signature. However, note that the axioms in best k -excerpts are not monotonic (cf. Example 112).

5.5 Evaluation

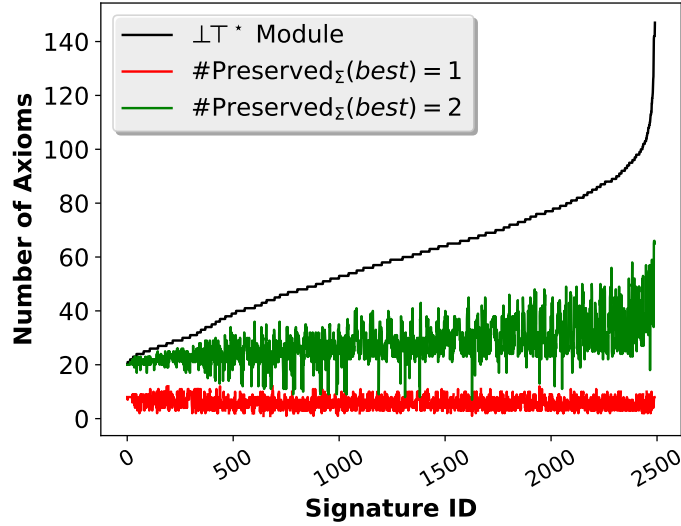
We have implemented our algorithms for computing best excerpts in Java. We used the prominent medical ontologies Snomed CT (version Jan 2016) consisting of 317891 axioms, and the NCI Thesaurus (NCI, version 16.03d) containing 165341 axioms.

Computation Time of Partial Max-SAT Encoding

The first experiment is to test the efficiency of our encoding method. Table 5.3 shows the time needed to compute best excerpts of NCI and Snomed CT for randomly generated signatures of different

Time (s)	Snomed CT				NCI			
	Minimal	Maximal	Median	STDEV	Minimal	Maximal	Median	STDEV
MaxSat	0.01	1.27	0.15	0.35	0.01	0.16	0.01	0.02

Table 5.3 Times for computing best excerpts encoding the input for and running SAT4J

Figure 5.2 Comparison of the size of best excerpts preserving one or two concept names with the size of $\perp T^*$ -local modules for 2500 signatures

sizes (10/30/50 concept names and 10/30/50 roles names). In total over 3000 signatures have been considered. After precomputing the subsumption justifications for every concept name in every signature, the time was measured as that needed to prepare the input and run the partial Max-SAT solver to compute best k -excerpts for every size k . The minimal, maximal, median, and standard deviation of the execution times show that computing best excerpts of any size using Max-SAT is very efficient given that the subsumption justifications have been computed.

Performance of Best Excerpts

In the second experiment, we compare the size of locality based modules with the number of axioms in best excerpts needed to preserve a certain amount of knowledge. We denote with $\#(\text{Preserved}_\Sigma(\text{best}) = n)$, for $n \in \{1, 2\}$, the minimal number of axioms needed by a best excerpt to preserve the knowledge of n concept names w.r.t. the signature Σ (i.e., the number k of axioms of a best excerpt \mathcal{E} of \mathcal{T} such that $n = |\Sigma \cap N_C \setminus cWtn_\Sigma(\mathcal{T}, \mathcal{E})|$). Instead of using random signatures, however, we consider a scenario where a user searches for sub-ontologies of Snomed CT related to a particular concept name. We computed 2500 different signatures each consisting of a concept name related to diseases, the TOP-concept (called ‘SNOMED CT Concept’) and all role names of Snomed CT. All weights of Σ -symbols are equal in this experiment.

In Figure 5.2, the 2500 signatures are presented along the x-axis in ascending order w.r.t. the size of their respective $\perp\top^*$ -local modules (black line). The y-axis represents the number of axioms in the module and excerpts for a signature. The red (resp. green) line presents the sizes of best excerpts that preserve the knowledge for one (resp. two) concept name(s), i.e., $\#Preserved_{\Sigma}(best) = 1$ (resp. $\#Preserved_{\Sigma}(best) = 2$). Best excerpts provide a conciser way for zooming in on an ontology, in particular, when sacrificing completeness is acceptable. For this experiment, after computing the subsumption justifications of all concept names in a signature, it only took 0.15s on average to compute the best excerpts.

Performance Comparison between Best Excerpts and IR Excerpts

In the third experiment, we compare the performance of best excerpts with IR excerpts [9] using the same set of signatures as in Fig. 5.2.

	Time (s)	#Axioms in excerpts	
		#Preserved $_{\Sigma}=1$	#Preserved $_{\Sigma}=2$
Best excerpts	94.20	5.90	28.34
IR excerpts	0.01	68.83	112.83

Table 5.4 Performance comparison of best and IR excerpts

Table 5.4 shows the average values of time and minimal size of best excerpts and IR excerpts when $\#Preserved_{\Sigma}$ is 1 and 2, respectively. Note that the computation time of best excerpts in Table 5.4 includes the computation time of subsumption justifications. Table 5.3 already shows that our encoding method is very efficient. Most of the computation time of best excerpts is used to compute subsumption justifications. We observed that it took only 0.01s to compute IR excerpts, which is much faster than computing best excerpts. However, for preserving the knowledge completeness $\#Preserved_{\Sigma} = 1$ or 2, the average minimal size of best excerpt is much less than IR excerpts. In other words, best excerpt is more succinct compared with IR excerpt w.r.t. the knowledge in signature.

Chapter 6

Extraction of Ontology Projection Module

Ontology comparison helps understanding the overlap and differences among ontologies, which is often desired while a user manipulates multiple knowledge sources. To understand the relations among these ontologies, in this chapter, we propose a novel notion called *projection module* that entails all queries that follow from a reference ontology. The aim is to study the techniques to extract *projection modules* which allow us to compare the reasoning capacities of different ontologies.

We develop a unified algorithm for computing minimal projection modules of acyclic \mathcal{ELH}^r -terminologies for subsumption, instance and conjunctive queries. The algorithms are based on simulation notions developed for detecting logical differences between \mathcal{ELH}^r -terminologies. Moreover, we prove that the relevant theoretic results are correct. Finally, we do some experimental evaluations to show that there exist projection modules between large biomedical ontologies, which shows the overlap and differences existing among different real-world ontologies.

6.1 Projection Modules

A projection module of an ontology gives a way to explain how the knowledge that is encoded in a reference ontology is implemented in the target ontology. We are interested in computing *all* projection modules, since it provides a complete list of all implementations of an ontology regarding a reference, each of which may be necessary to be checked. To enable a manual validation by domain experts, we need to present only necessary information, so we focus on computing *minimal* projection modules.

A terminology \mathcal{T}_1 together with a signature Σ and a query language Q determine a set Φ of queries from Q formulated using only symbols from Σ that follow from \mathcal{T}_1 . A projection module of another terminology \mathcal{T}_2 is a subset of \mathcal{T}_2 that entails the queries in Φ . For convenience, we bundle the parameters together in a tuple $\rho = \langle \mathcal{T}_1, \Sigma, \mathcal{T}_2 \rangle$, which we call a *projection setting*.

Definition 137 (Projection Module) Let $\mathcal{T}_1, \mathcal{T}_2$ be two \mathcal{ELH}^r -terminologies and Σ a signature. Additionally, let $\rho = \langle \mathcal{T}_1, \Sigma, \mathcal{T}_2 \rangle$ be a projection setting, and \mathcal{A} be a Σ -ABox. A subset $\mathcal{M} \subseteq \mathcal{T}_2$ is a subsumption (resp. instance, conjunctive query) projection module under projection setting ρ , denoted as \mathcal{M}_ρ^c (resp. $\mathcal{M}_\rho^i, \mathcal{M}_\rho^q$) if:

- \mathcal{M}_ρ^c : for each \mathcal{ELH}_Σ^r -inclusion α , $\mathcal{T}_1 \models \alpha$ implies $\mathcal{M} \models \alpha$;
- \mathcal{M}_ρ^i : for each Σ -instance assertion λ , $(\mathcal{T}_1, \mathcal{A}) \models \lambda$ implies $(\mathcal{M}, \mathcal{A}) \models \lambda$;
- \mathcal{M}_ρ^q : for each $q(\vec{a})$, $(\mathcal{T}_1, \mathcal{A}) \models q(\vec{a})$ implies $(\mathcal{M}, \mathcal{A}) \models q(\vec{a})$, where \vec{a} is a tuple of individual names in \mathcal{A} and $q(\vec{a})$ is a Σ -conjunctive query.

A minimal subsumption (resp. instance, query) projection module is a projection module \mathcal{M}_ρ^c (resp. $\mathcal{M}_\rho^i, \mathcal{M}_\rho^q$) minimal w.r.t. \subseteq .

Projection module is a notion more fine-grained than logical difference in the sense that when an ontology \mathcal{T}_1 has no logical difference with the other \mathcal{T}_2 , we can further extract sub-ontologies of \mathcal{T}_1 that contain the knowledge in the reference ontology \mathcal{T}_2 .

Example 138 Let $\mathcal{T}_1 = \{A_1 \sqsubseteq A_2, A_2 \sqsubseteq A_3\}$, $\mathcal{T}_2 = \{A_1 \sqsubseteq A_3 \sqcap B_1, B_1 \sqsubseteq \exists r.A_3\}$, and $\Sigma = \{A_1, A_3, r\}$. Additionally, let $\rho = \langle \mathcal{T}_1, \Sigma, \mathcal{T}_2 \rangle$. We have that $\text{cDiff}_\Sigma(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$. However, $\mathcal{M}_\rho^c = \{A_1 \sqsubseteq A_3 \sqcap B_1\}$, So we have that $\mathcal{M}_\rho^c \subsetneq \mathcal{T}_2$, which means that a strict sub-ontology of \mathcal{T}_2 is sufficient to capture all concept subsumption information of \mathcal{T}_1 about Σ . Moreover, \mathcal{T}_2 also entails a consequence $A_1 \sqsubseteq \exists r.A_3$, which is not the case for \mathcal{T}_1 .

The following example shows that the three notions of projection modules based on different query languages are distinct.

Example 139 Let $\Sigma = \{X, Y, Z, B, r, s\}$ and $\mathcal{T} = \{X \sqsubseteq Y, Y \sqsubseteq \exists t.Z, \text{ran}(r) \sqsubseteq A_1, \text{ran}(s) \sqsubseteq A_2, B \equiv A_1 \sqcap A_2\}$. Additionally, let $\rho = \langle \mathcal{T}, \Sigma, \mathcal{T} \rangle$. We have that $\mathcal{M}_\rho^c = \{X \sqsubseteq Y\}$, $\mathcal{M}_\rho^i = \mathcal{M}_\rho^c \cup \{\text{ran}(r) \sqsubseteq A_1, \text{ran}(s) \sqsubseteq A_2, B \equiv A_1 \sqcap A_2\}$ and $\mathcal{M}_\rho^q = \mathcal{T}$.

Note that there might exist several, even exponentially many, subsumption projection modules given a reference ontology, an implement ontology, and a signature.

Example 140 Let $\mathcal{T}_1 = \{X \sqsubseteq Z\}$ and $\mathcal{T}_2 = \{X \sqsubseteq Y_1 \sqcap Y_2, Y_1 \sqsubseteq Z, Y_2 \sqsubseteq Z\}$ and $\Sigma = \{X, Z\}$. Additionally, let $\rho = \langle \mathcal{T}_1, \Sigma, \mathcal{T}_2 \rangle$. We have that both $\{X \sqsubseteq Y_1 \sqcap Y_2, Y_1 \sqsubseteq Z\}$ and $\{X \sqsubseteq Y_1 \sqcap Y_2, Y_2 \sqsubseteq Z\}$ are subsumption projection modules under $\rho = \langle \mathcal{T}_1, \Sigma, \mathcal{T}_2 \rangle$.

Definition 141 (Q-Projection Simulation) Let $\mathcal{T}_1, \mathcal{T}_2$ be two \mathcal{ELH}^r -terminologies and Σ be a signature. Additionally, let $Q \in \{c, i, q\}$. We say that \mathcal{T}_1 is Q -projection simulated by \mathcal{T}_2 under the setting $\rho = \langle \mathcal{T}_1, \Sigma, \mathcal{T}_2 \rangle$ (written $\mathcal{T}_1 \rightsquigarrow_\Sigma^Q \mathcal{T}_2$) iff there exists a Q -projection module $\mathcal{M}_\rho^Q \subseteq \mathcal{T}_2$ under ρ .

Example 142 Let $\mathcal{T}_1 = \{X \sqsubseteq Z\}$ and $\mathcal{T}_2 = \{X \sqsubseteq Y, Y \sqsubseteq Z\}$ and $\Sigma = \{X, Z\}$. Additionally, let $\rho = \langle \mathcal{T}_1, \Sigma, \mathcal{T}_2 \rangle$. We have that $\mathcal{M}_\rho^Q = \mathcal{T}_2$ for $Q \in \{c, i, q\}$. So $\mathcal{T}_1 \rightsquigarrow_\Sigma^Q \mathcal{T}_2$.

The Q -projection relation satisfies the monotonicity properties as described below:

Proposition 143 Let $\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3$ be \mathcal{ELH}^r -terminologies, Σ be a signature and $Q \in \{c, i, q\}$. If $\mathcal{T}_2 \subseteq \mathcal{T}_3$ and $\mathcal{T}_1 \rightsquigarrow_\Sigma^Q \mathcal{T}_2$, then $\mathcal{T}_1 \rightsquigarrow_\Sigma^Q \mathcal{T}_3$.

This proposition shows that if we extend the ontology from \mathcal{T}_2 to \mathcal{T}_3 , the larger ontology \mathcal{T}_3 will Q -projection simulate all the ontologies that \mathcal{T}_2 Q -projection simulates.

Proposition 144 Let $\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3$ be \mathcal{ELH}^r -terminologies, Σ be a signature and $Q \in \{c, i, q\}$. If $\mathcal{T}_1 \subseteq \mathcal{T}_2$ and $\mathcal{T}_2 \rightsquigarrow_\Sigma^Q \mathcal{T}_3$ then $\mathcal{T}_1 \rightsquigarrow_\Sigma^Q \mathcal{T}_3$.

By this proposition, if an ontology \mathcal{T}_3 Q -projection simulates \mathcal{T}_1 , then \mathcal{T}_3 also Q -projection simulates all sub-ontologies of \mathcal{T}_1 .

Now we consider projection modules under a special setting of the form $\rho^\circ = \langle \mathcal{T}, \Sigma, \mathcal{T} \rangle$, that is, the reference ontology is also the implementing ontology (Example 139). A projection module under ρ° is also called an *automorphic projection module*.

Proposition 145 Let \mathcal{T} be an \mathcal{ELH}^r -terminology, Σ a signature and $\rho^\circ = \langle \mathcal{T}, \Sigma, \mathcal{T} \rangle$. Additionally, let $\mathcal{M}_{\rho^\circ}^c$ be a subsumption projection module under ρ° . Then $\mathcal{M}_{\rho^\circ}^c$ is a basic \mathcal{ELH}^r -subsumption module of \mathcal{T} w.r.t. Σ .

This proposition can be obtained from Definition 93 and Definition 137. It shows that we can compute basic \mathcal{ELH}^r -subsumption modules by computing automorphic projection modules.

In Example 138, the automorphic projection module of \mathcal{T}_2 is \mathcal{T}_2 itself, which is different from its projection module w.r.t. \mathcal{T}_1 as discussed above. It shows that automorphic projection module is indeed a different notion from projection module.

6.2 Computing Role Projection Justifications

Definition 146 (Role Projection Justification) Let \mathcal{T}_1 and \mathcal{T}_2 be \mathcal{ELH}^r -terminologies and Σ be a signature. Additionally, let $\rho = \langle \mathcal{T}_1, \Sigma, \mathcal{T}_2 \rangle$. A set $\mathcal{M} \subseteq \mathcal{T}_2$ is called a role projection module under ρ , denoted by \mathcal{J}_ρ^R , iff for every $s \in \Sigma \cap \mathbf{N}_R$, $\mathcal{T}_1 \models r_1 \sqsubseteq s$ implies $\mathcal{M} \models r_2 \sqsubseteq s$. A role projection justification under ρ is the role subsumption module under ρ that is minimal w.r.t. \subseteq .

We denote the set of all role projection justifications under ρ as \mathbb{J}_ρ^R .

Example 147 Let $\mathcal{T}_1 = \{r \sqsubseteq t\}$ and $\mathcal{T}_2 = \{r \sqsubseteq s, s \sqsubseteq t\}$ and $\Sigma = \{r, t\}$. Additionally, let $\rho = \langle \mathcal{T}_1, \Sigma, \mathcal{T}_2 \rangle$. We have that $\mathbb{J}_\rho^R = \{\mathcal{T}_2\} = \{\{r \sqsubseteq s, s \sqsubseteq t\}\}$.

Algorithm 18: Computing the set of all Projection Justifications for Role Inclusions

```

1 function COVERR ( $\mathcal{T}_1, \Sigma, \mathcal{T}_2$ )
2    $\mathbb{M} = \{\emptyset\}$ 
3   for every  $r, s \in \Sigma \cap \mathbb{N}_R$  such that  $\mathcal{T}_1 \models r \sqsubseteq s$  do
4      $\mathbb{M} := \mathbb{M} \otimes \text{Just}_{\mathcal{T}_2}(r \sqsubseteq s)$ 
5   end
6   return Minimise $\sqsubseteq$ ( $\mathbb{M}$ )

```

Lemma 148 Let \mathcal{T}_1 and \mathcal{T}_2 be \mathcal{ELH}^r -terminologies and Σ be a signature. Additionally, let $\rho = \langle \mathcal{T}_1, \Sigma, \mathcal{T}_2 \rangle$. If $\mathcal{I}_R \in \mathbb{J}_\rho^R$, then $\text{roleWtn}_\Sigma(\mathcal{T}_1, \mathcal{I}_R) = \emptyset$.

Algorithm 18 shows how to collect relevant Σ -role inclusions by Definition 146. The set of all role projection justifications of an \mathcal{ELH}^r -terminology can therefore be computed in polynomial time.

Theorem 149 Let \mathcal{T}_1 and \mathcal{T}_2 be \mathcal{ELH}^r -terminologies and Σ be a signature. Additionally, let $\rho = \langle \mathcal{T}_1, \Sigma, \mathcal{T}_2 \rangle$. Let $\mathbb{M} := \text{COVER}^R(\mathcal{T}_1, \Sigma, \mathcal{T}_2)$ computed by Algorithm 18. Then \mathbb{M} is the set of all role projection modules under ρ .

This theorem follows from Definition 47 and Definition 146.

6.3 Computing Subsumption Projection Justifications

The notion of projection justification between terminologies depends on several parameters. To better manage these parameters, we wrap them in a *subsumption setting* χ , which is a tuple of the form $\langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L} \rangle$, where \mathcal{T}_1 and \mathcal{T}_2 are the normalised \mathcal{ELH}^r -terminologies, Σ is a signature, $X_1, X_2 \in \mathbb{N}_C \cup \{\text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_R\}$, and $\mathcal{L} \in \{\mathcal{ELH}^r, \mathcal{EL}^{\text{ran}}, \mathcal{EL}^{\text{ran}, \square, u}\}$. We recall that ran , Ran and $\text{Ran}^{\square, u}$ are abbreviations for \mathcal{ELH}^r , $\mathcal{EL}^{\text{ran}}$ and $\mathcal{EL}^{\text{ran}, \square, u}$, respectively. Besides, \mathcal{L}_Σ means \mathcal{L} -concept that only consists of the concept and role names in Σ .

Definition 150 (Subsumption Projection Justification) Let $\mathcal{L} \in \{\text{ran}, \text{Ran}, \text{Ran}^{\square, u}\}$. Additionally, let $\chi = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L} \rangle$, where $X_1, X_2 \in \mathbb{N}_C \cup \{\text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_R\}$. A set $\mathcal{M} \subseteq \mathcal{T}_2$ is called a *subsumer projection module* under χ iff for every $C \in \mathcal{L}_\Sigma$, $\mathcal{T}_1 \models X_1 \sqsubseteq C$ implies $\mathcal{M} \models X_2 \sqsubseteq C$. $\mathcal{M} \subseteq \mathcal{T}_2$ is called a *subsumee projection module* under χ iff for every $C \in \mathcal{L}_\Sigma$, $\mathcal{T}_1 \models C \sqsubseteq X_1$ implies $\mathcal{M} \models C \sqsubseteq X_2$.

\mathcal{M} is called a *subsumption projection module* under χ iff \mathcal{M} is a *subsumer projection module* and a *subsumee projection module* under χ . A subsumee (resp. subsumer, subsumption) projection justification under χ is a subsumee (resp. subsumer, subsumption) projection projection module under χ that is minimal w.r.t. \sqsubseteq .

We denote the set of all subsumee (resp. subsumer, subsumption) projection justifications under χ as $\mathbb{J}_{\chi}^{\leftarrow}$ (resp. $\mathbb{J}_{\chi}^{\rightarrow}$, \mathbb{J}_{χ}), where $\chi = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L} \rangle$.

Proposition 151 *Let \mathcal{T} be a normalised \mathcal{ELH}^r -terminology, let Σ be a signature, and let $X \in \mathbb{N}_{\mathbb{C}}$. Additionally, let $\chi = \langle \mathcal{T}, X, \Sigma, \mathcal{T}, X, \text{ran} \rangle$. Then a subsumer (resp. subsumee, subsumption) projection module under χ is a $\langle X, \Sigma \rangle$ -subsumer (resp. subsumee, subsumption) module. Similarly, a subsumer (resp. subsumee, subsumption) projection justification under χ is a $\langle X, \Sigma \rangle$ -subsumer (resp. subsumee, subsumption) justification.*

This proposition follows from Definition 79 and Definition 137.

Using Definition 93 and Definition 19, we obtain the following lemma stating the absence of certain concept names, and domain and range restrictions of role names as left-hand and right-hand difference witnesses between a reference terminology \mathcal{T}_1 and a subsumer and subsumee justification of a second terminology \mathcal{T}_2 .

Lemma 152 *Let $\chi(\varphi) = \langle \mathcal{T}_1, \varphi, \Sigma, \mathcal{T}_2, \varphi, \mathcal{L} \rangle$ and $\chi(A) = \langle \mathcal{T}_1, A, \Sigma, \mathcal{T}_2, A, \mathcal{L} \rangle$, where $\varphi \in (\Sigma \cap \mathbb{N}_{\mathbb{C}}) \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}$, $A \in \Sigma \cap \mathbb{N}_{\mathbb{C}}$ and $\mathcal{L} \in \{\text{ran}, \text{Ran}, \text{Ran}^{\square, u}\}$. Then we have that for every $\mathcal{I}_{\chi(\varphi)} \in \mathbb{J}_{\chi(\varphi)}^{\rightarrow}$, $\varphi \notin \text{lhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{I}_{\chi(\varphi)})$, and for every $\mathcal{I}_{\chi(A)} \in \mathbb{J}_{\chi(A)}^{\leftarrow}$, $A \notin \text{rhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{I}_{\chi(A)})$.*

To obtain subsumption modules, we can use the operator \otimes to combine sets of role projection justifications, subsumer and subsumee projection justifications, one justification for each potential difference witness that needs to be prevented; cf. Lemma 152 and 148. This is made precise in the following theorem.

Theorem 153 *Let $\mathcal{L} \in \{\text{ran}, \text{Ran}, \text{Ran}^{\square, u}\}$ and $\mathbb{M}_{\rho}^{\mathcal{L}}$ be the set of all \mathcal{L} -projection modules under $\rho = \langle \mathcal{T}_1, \Sigma, \mathcal{T}_2 \rangle$ that are minimal w.r.t. \sqsubset . Then*

$$\mathbb{M}_{\rho}^{\mathcal{L}} = \text{Minimize}_{\sqsubset} \left(\mathbb{J}_{\rho}^{\text{R}} \otimes \bigotimes_{\varphi \in (\Sigma \cap \mathbb{N}_{\mathbb{C}}) \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}} \mathbb{J}_{\chi(\varphi)}^{\rightarrow} \otimes \bigotimes_{A \in \Sigma \cap \mathbb{N}_{\mathbb{C}}} \mathbb{J}_{\chi(A)}^{\leftarrow} \right)$$

where $\chi(\psi) = \langle \mathcal{T}_1, \psi, \Sigma, \mathcal{T}_2, \psi, \mathcal{L} \rangle$.

In [23], it is shown that concept difference on $\mathcal{EL}^{\text{ran}}$ and $\mathcal{EL}^{\text{ran}, \square, u}$ captures exactly the instance and query difference in \mathcal{ELH}^r -terminology, respectively. So we can convert the problem of computing instance and conjunctive query projection modules to the problem of computing subsumption projection modules on $\mathcal{EL}^{\text{ran}}(\text{Ran})$ and $\mathcal{EL}^{\text{ran}, \square, u}(\text{Ran}^{\square, u})$ respectively. Therefore, in the following subsections, we present a general algorithm for computing subsumer and subsumee projection justifications for subsumption, conjunctive and instance query by three different logics: $\mathcal{ELH}^r(\text{ran})$, $\mathcal{EL}^{\text{ran}}(\text{Ran})$ and $\mathcal{EL}^{\text{ran}, \square, u}(\text{Ran}^{\square, u})$.

Due to the fact that there may exist exponentially many minimal modules in the worst case, our algorithm introduced later runs in exponential-time in the worst case.

As in Chapter 3, we use the notion of covers to characterise the output of our algorithms to ensure that all justifications have been computed. We recall that \mathcal{L}_{Σ} is a \mathcal{L} concept that only constructed by the concept and role names in Σ .

6.3.1 Computing Subsumer Projection Justifications

In this section, we introduce our algorithm to compute all subsumer projection justifications. The algorithm relies on the notion of a subsumer simulation between terminologies.

In [23] it was shown that

- $\text{lhsWtn}_{\Sigma}^{\text{ran}}(\mathcal{T}_1, \mathcal{T}_2) = \text{lhsWtn}_{\Sigma}^{\text{Ran}}(\mathcal{T}_1, \mathcal{T}_2)$;
- $\text{lhsWtn}_{\Sigma}^{\text{Ran}}(\mathcal{T}_1, \mathcal{T}_2) \neq \text{lhsWtn}_{\Sigma}^{\text{Ran}^{\square, u}}(\mathcal{T}_1, \mathcal{T}_2)$.

Therefore, we just need algorithms to deal with two query languages, i.e., Ran and $\text{Ran}^{\square, u}$.

To handle existential restrictions via the universal role in the query language $\text{Ran}^{\square, u}$, we use a notion of reachability in terminologies. Let $\text{N}_{\mathcal{C}}^{\text{dom, ran}} = \text{N}_{\mathcal{C}} \cup \{\text{dom}(r) \mid r \in \text{N}_{\mathcal{R}}\} \cup \{\text{ran}(r) \mid r \in \text{N}_{\mathcal{R}}\}$. For $\varphi \in \text{N}_{\mathcal{C}}^{\text{dom, ran}}$ and a normalised $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminology \mathcal{T} , let $F_{\mathcal{T}}(\varphi) \subseteq \text{N}_{\mathcal{C}}^{\text{dom, ran}}$ be the smallest set closed under the following conditions:

- $\varphi \in F_{\mathcal{T}}(\varphi)$;
- $Y \in F_{\mathcal{T}}(\varphi)$ if $\psi \in F_{\mathcal{T}}(\varphi)$, $\mathcal{T} \models \psi \sqsubseteq X'$ and $X' \bowtie \exists r.Y \in \mathcal{T}$;
- $\text{dom}(r) \in F_{\mathcal{T}}(\varphi)$ if $\text{ran}(r) \in F_{\mathcal{T}}(\varphi)$.

Intuitively, for concept names X , the set $F_{\mathcal{T}}(X)$ contains X together with all directly and indirectly implied role successors of X , independent of any signature. For example, let $\mathcal{T} = \{Y \sqsubseteq X, X \equiv \exists r.Z, Z \sqsubseteq \exists r.B\}$. Then it holds that $F_{\mathcal{T}}(Y) = \{Y, Z, B\}$, $F_{\mathcal{T}}(X) = \{X, Z, B\}$, $F_{\mathcal{T}}(Z) = \{Z, B\}$, $F_{\mathcal{T}}(B) = \{B\}$, $F_{\mathcal{T}}(\text{dom}(r)) = \{\text{dom}(r)\}$ and $F_{\mathcal{T}}(\text{ran}(r)) = \{\text{dom}(r), \text{ran}(r)\}$.

In order to deal with $\text{Ran}^{\square, u}$ -subsumption queries, we extend Definition 83 and get the following definition.

Definition 154 (Subsumer Simulation for Ran and $\text{Ran}^{\square, u}$ Subsumption Query) *Let \mathcal{T}_1 and \mathcal{T}_2 be two normalised $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies, and let Σ be a signature. Moreover, let $\mathcal{L} \in \{\text{Ran}, \text{Ran}^{\square, u}\}$ and let $N(\mathcal{T}, \Sigma) = \{X, \text{dom}(r), \text{ran}(r) \mid X, r \in (\text{sig}(\mathcal{T}) \cup \Sigma), X \in \text{N}_{\mathcal{C}}, r \in \text{N}_{\mathcal{R}}\}$, where \mathcal{T} ranges over terminologies. A relation $S \subseteq N(\mathcal{T}_1, \Sigma) \times N(\mathcal{T}_2, \Sigma)$ is a $\langle \Sigma, \mathcal{L} \rangle$ -subsumer simulation from \mathcal{T}_1 to \mathcal{T}_2 iff the following conditions $(S_{\text{N}_{\mathcal{C}}}^{\rightarrow})$ and $(S_{\exists, \mathcal{L}}^{\rightarrow})$ are both satisfied:*

$(S_{\text{N}_{\mathcal{C}}}^{\rightarrow})$ if $(X_1, X_2) \in S$, then for every $\varphi \in \Sigma \cup \Sigma^{\text{dom}}$ with $\mathcal{T}_1 \models X_1 \sqsubseteq \varphi$, it holds that $\mathcal{T}_2 \models X_2 \sqsubseteq \varphi$;

$(S_{\exists, \text{Ran}}^{\rightarrow})$ if $(X_1, X_2) \in S$ and $X_1' \bowtie_1 \exists r.Y_1 \in \mathcal{T}_1$ with $\bowtie_1 \in \{\sqsubseteq, \equiv\}$ such that $\mathcal{T}_1 \models X_1 \sqsubseteq X_1'$ and $\mathcal{T}_1 \models r \sqsubseteq s$ for some $s \in \Sigma$, then for every $s \in \Sigma$ with $\mathcal{T}_1 \models r \sqsubseteq s$, there exists $X_2' \bowtie_2 \exists r'.Y_2 \in \mathcal{T}_2$ with $\bowtie_2 \in \{\sqsubseteq, \equiv\}$ such that $\mathcal{T}_2 \models X_2 \sqsubseteq X_2'$, it holds that $\mathcal{T}_2 \models r' \sqsubseteq s$ and $(Y_1, Y_2) \in S$;

$(S_{\exists, \text{Ran}^{\square, u}}^{\rightarrow})$ if $(X_1, X_2) \in S$ and $X_1' \bowtie_1 \exists r.Y_1 \in \mathcal{T}_1$ with $\bowtie_1 \in \{\sqsubseteq, \equiv\}$ such that $\mathcal{T}_1 \models X_1 \sqsubseteq X_1'$ and $\mathcal{T}_1 \models r \sqsubseteq s$ for some $s \in \Sigma$, there exists $X_2' \bowtie_2 \exists r'.Y_2 \in \mathcal{T}_2$ with $\bowtie_2 \in \{\sqsubseteq, \equiv\}$ such that $\mathcal{T}_2 \models X_2 \sqsubseteq X_2'$ and for every $s \in \Sigma$ with $\mathcal{T}_1 \models r \sqsubseteq s$, it holds that $\mathcal{T}_2 \models r' \sqsubseteq s$ and $(Y_1, Y_2) \in S$.

We write $\mathcal{T}_1 \sim_{\Sigma, \text{Ran}}^{\rightarrow} \mathcal{T}_2$ iff there exists a $\langle \Sigma, \text{Ran} \rangle$ -subsumer simulation S from \mathcal{T}_1 to \mathcal{T}_2 such that for every $\varphi \in (\Sigma \cap \text{N}_C) \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}$: $(\varphi, \varphi) \in S$.

Moreover, we write $\mathcal{T}_1 \sim_{\Sigma, \text{Ran}^{\square, u}}^{\rightarrow} \mathcal{T}_2$ iff there exists a $\langle \Sigma, \text{Ran}^{\square, u} \rangle$ -subsumer simulation S from \mathcal{T}_1 to \mathcal{T}_2 such that for every $\varphi \in (\Sigma \cap \text{N}_C) \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}$: $(\varphi, \varphi) \in S$, and for every $\psi \in F_{\mathcal{T}_1}(\varphi)$, there exists a $\psi' \in F_{\mathcal{T}_2}(\psi)$ such that $(\psi, \psi') \in S$.

For $X_1, X_2 \in \text{N}_C$ and $\mathcal{L} \in \{\text{Ran}, \text{Ran}^{\square, u}\}$, we write $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{L}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$ iff there exists a $\langle \Sigma, \mathcal{L} \rangle$ -subsumer simulation S from \mathcal{T}_1 to \mathcal{T}_2 with $(X_1, X_2) \in S$ for which $\mathcal{T}_1 \sim_{\Sigma, \mathcal{L}}^{\rightarrow} \mathcal{T}_2$.

This definition of subsumer simulation is extended from Definition 83 to additionally handle $\text{Ran}^{\square, u}$ -subsumption queries.

A subsumer simulation conveniently captures the set of subsumers in the following sense: If a Σ -subsumer simulation from \mathcal{T}_1 to \mathcal{T}_2 contains the pair (X_1, X_2) , then X_2 entails w.r.t. \mathcal{T}_2 all subsumers of X_1 w.r.t. \mathcal{T}_1 that are formulated in the signature Σ . Formally, we obtain the following theorem from [34].

Theorem 155 *We have the following conclusions:*

- $\mathcal{T}_1 \sim_{\Sigma, \text{ran}}^{\rightarrow} \mathcal{T}_2$ iff $\text{lhsWtn}_{\Sigma}^{\text{ran}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$;
- $\mathcal{T}_1 \sim_{\Sigma, \text{Ran}}^{\rightarrow} \mathcal{T}_2$ iff $\text{lhsWtn}_{\Sigma}^{\text{Ran}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$;
- $\mathcal{T}_1 \sim_{\Sigma, \text{Ran}^{\square, u}}^{\rightarrow} \mathcal{T}_2$ iff $\text{lhsWtn}_{\Sigma}^{\text{Ran}^{\square, u}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$.

Theorem 156 *If $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{L}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$, then $\mathcal{T}_2 \models X_2 \sqsubseteq D$ for every $D \in \mathcal{L}_{\Sigma}$ such that $\mathcal{T}_1 \models X_1 \sqsubseteq D$.*

Note that $\mathcal{T}_2 \models X_2 \sqsubseteq D$ for every $D \in \mathcal{L}_{\Sigma}$ such that $\mathcal{T}_1 \models X_1 \sqsubseteq D$ does not imply that $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{L}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$, because $\mathcal{T}_1 \sim_{\Sigma, \mathcal{L}}^{\rightarrow} \mathcal{T}_2$ is the one of conditions that $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{L}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$ holds (cf. Definition 154).

For example, let $\mathcal{T}_1 = \{X_1 \sqsubseteq A, B \sqsubseteq B'\}$, $\mathcal{T}_2 = \{X_2 \sqsubseteq A\}$ and $\Sigma = \{A, B, B'\}$. Then $B \notin F_{\mathcal{T}_1}(X_1)$. Note that $\mathcal{T}_1 \sim_{\Sigma, \mathcal{L}}^{\rightarrow} \mathcal{T}_2$ does not hold as B in \mathcal{T}_1 is not simulated by anything in \mathcal{T}_2 . Therefore, also $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{L}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$ does not hold. However, we have that for every $D \in \mathcal{L}_{\Sigma}$: $\mathcal{T}_1 \models X_1 \sqsubseteq D$ implies $\mathcal{T}_2 \models X_2 \sqsubseteq D$. In particular, $\mathcal{T}_1 \models X_1 \sqsubseteq A$ and $\mathcal{T}_2 \models X_2 \sqsubseteq A$.

When computing subsumer projection justification, in Algorithm 26, it distinguishes different cases. If $\mathcal{L} = \text{ran}$ or Ran , then the algorithm calls Algorithm 23 and Algorithm 22.

In order to collect the axioms necessary to satisfy the conditions of Definition 154, we develop Algorithm 19 for Case $(S_{\text{N}_C}^{\rightarrow})$, Algorithm 20 for Case $(S_{\exists, \text{Ran}}^{\rightarrow})$, and Algorithm 21 for Case $(S_{\exists, \text{Ran}^{\square, u}}^{\rightarrow})$. Note that Algorithm 19 is same for ran -, Ran - and $\text{Ran}^{\square, u}$ -subsumption queries.

Observe that $\text{COVER}_{\rightarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L})$ may be called several times during the execution of Algorithm 22. The algorithm can be optimised by caching the return value of the first execution, and retrieving it from memory for subsequent calls.

Algorithm 19: Computing the set of all Subsumer Projection Justifications ($S_{N_C}^{\rightarrow}$)

```

1 function COVER $_{\rightarrow}^{N_C}$  ( $\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2$ )
2    $M_{X_2}^{\rightarrow} = \{\emptyset\}$ 
3   for every  $B \in (\Sigma \cap N_C) \cup \{\text{dom}(r) \mid r \in \Sigma\}$  such that  $\mathcal{T}_1 \models X_1 \sqsubseteq B$  do
4      $M_{X_2}^{\rightarrow} := M_{X_2}^{\rightarrow} \otimes \text{Just}_{\mathcal{T}_2}(X_2 \sqsubseteq B)$ 
5   end
6   return Minimise $_{\subseteq}(M_{X_2}^{\rightarrow})$ 

```

Algorithm 20: Computing the set of all Subsumer Projection Justifications ($S_{\exists, \text{Ran}}^{\rightarrow}$) ($\mathcal{L} \in \{\text{ran}, \text{Ran}\}$)

```

1 function COVER $_{\rightarrow}^{1, \exists}$  ( $\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L}$ )
2    $M_X^{\rightarrow} = \{\emptyset\}$ 
3   for every  $Y \bowtie_1 \exists r.Z \in \mathcal{T}_1$  ( $\bowtie_1 \in \{\sqsubseteq, \equiv\}$ ) and  $s \in \Sigma \cap N_R$  such that  $\mathcal{T}_1 \models X \sqsubseteq Y$  and
    $\mathcal{T}_1 \models r \sqsubseteq s$  do
4      $M_{\exists s.Z}^{\rightarrow} := \{\emptyset\}$ 
5     for every  $Y' \bowtie_2 \exists r'.Z' \in \mathcal{T}_2$  ( $\bowtie_2 \in \{\sqsubseteq, \equiv\}$ ) such that  $\mathcal{T}_2 \models X \sqsubseteq Y'$ ,  $\mathcal{T}_2 \models r' \sqsubseteq s$ , and
      $\langle \mathcal{T}_1, z \rangle \sim_{\Sigma, \mathcal{L}}^{\rightarrow} \langle \mathcal{T}_2, z' \rangle$  do
6        $M_{Z'}^{\rightarrow} := \text{COVER}_{\rightarrow}(\mathcal{T}_1, Z, \Sigma, \mathcal{T}_2, Z', \mathcal{L})$ 
7        $M_{\exists s.Z}^{\rightarrow} := M_{\exists s.Z}^{\rightarrow} \cup (\{\{Y' \bowtie_2 \exists r'.Z'\}\} \otimes M_{Z'}^{\rightarrow} \otimes \text{Just}_{\mathcal{T}_2}(X \sqsubseteq Y') \otimes \text{Just}_{\mathcal{T}_2}(r' \sqsubseteq s))$ 
8     end
9      $M_X^{\rightarrow} := M_X^{\rightarrow} \otimes M_{\exists s.Z}^{\rightarrow}$ 
10  end
11  return Minimise $_{\subseteq}(M_X^{\rightarrow})$ 

```

The following theorem shows that Algorithm 22 indeed computes the set of subsumer projection justifications.

Theorem 157 *Let \mathcal{T}_1 and \mathcal{T}_2 be acyclic, normalised \mathcal{ELH}^r -terminologies, and let Σ be a signature. Additionally, let $\chi = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L} \rangle$ and $\mathbb{M} := \text{COVER}_{\rightarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L})$ computed by Algorithm 22. If $\mathcal{T}_1 \sim_{\Sigma, \mathcal{L}}^{\rightarrow} \mathcal{T}_2$, then \mathbb{M} is the set of all subsumer projection justifications under χ .*

6.3.2 Computing Subsumee Projection Justifications

We now introduce algorithms of computing subsumee projection justifications based on subsumee simulation. The basic idea of the algorithm is to collect as few axioms from \mathcal{T}_2 as possible to maintain the subsumee simulation between Σ -concept names. In [23], we know that

- $\text{rhsWtn}_{\Sigma}^{\text{ran}}(\mathcal{T}_1, \mathcal{T}_2) \neq \text{rhsWtn}_{\Sigma}^{\text{Ran}}(\mathcal{T}_1, \mathcal{T}_2)$;
- $\text{rhsWtn}_{\Sigma}^{\text{Ran}}(\mathcal{T}_1, \mathcal{T}_2) = \text{rhsWtn}_{\Sigma}^{\text{Ran}^{\cap, \cup}}(\mathcal{T}_1, \mathcal{T}_2)$.

Algorithm 21: Computing the set of all Subsumer Projection Justifications ($S_{\exists, \text{Ran}^{\cap, u}}^{\rightarrow}$) ($\mathcal{L} = \text{Ran}^{\cap, u}$)

```

1 function COVER $_{\rightarrow}^{2, \exists}$  ( $\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L}$ )
2    $M_{(X_1, X_2)}^{\rightarrow} = \{\emptyset\}$ 
3   for every  $Y \bowtie_1 \exists r.Z \in \mathcal{T}_1$  ( $\bowtie_1 \in \{\sqsubseteq, \equiv\}$ ) with  $\mathcal{T}_1 \models X_1 \sqsubseteq Y$ ,  $\mathcal{T}_1 \models r \sqsubseteq s$  for some
4      $s \in \Sigma \cap N_R$  do
5      $M_{\exists r.Z}^{\rightarrow} := \{\emptyset\}$ 
6     for every  $Y' \bowtie_2 \exists r'.Z' \in \mathcal{T}_2$  ( $\bowtie_2 \in \{\sqsubseteq, \equiv\}$ ) with  $\mathcal{T}_2 \models X_2 \sqsubseteq Y'$  and  $\mathcal{T}_2 \models r' \sqsubseteq s$  for
7       every  $s \in \{s' \in \Sigma \cap N_R \mid \mathcal{T}_1 \models r \sqsubseteq s'\}$  and  $\langle \mathcal{T}_1, Z \rangle \sim_{\Sigma, \mathcal{L}}^{\rightarrow} \langle \mathcal{T}_2, Z' \rangle$  do
8        $M_{r'}^{\rightarrow} := \{\emptyset\}$ 
9       for every  $s \in \Sigma \cap N_R$  with  $\mathcal{T}_1 \models r \sqsubseteq s$  do
10         $M_{r'}^{\rightarrow} := M_{r'}^{\rightarrow} \otimes \text{Just}_{\mathcal{T}_2}(r' \sqsubseteq s)$ 
11      end
12       $M_{Z'}^{\rightarrow} := \text{COVER}_{\rightarrow}(\mathcal{T}_1, Z, \Sigma, \mathcal{T}_2, Z', \mathcal{L})$ 
13       $M_{\exists r.Z}^{\rightarrow} := M_{\exists r.Z}^{\rightarrow} \cup (\text{Just}_{\mathcal{T}_2}(X_2 \sqsubseteq Y') \otimes \{\{Y' \bowtie_2 \exists r'.Z'\}\} \otimes M_{r'}^{\rightarrow} \otimes M_{Z'}^{\rightarrow})$ 
14    end
15     $M_{(X_1, X_2)}^{\rightarrow} := M_{(X_1, X_2)}^{\rightarrow} \otimes M_{\exists r.Z}^{\rightarrow}$ 
16 end
17 return Minimise $_{\subseteq}(M_{(X_1, X_2)}^{\rightarrow})$ 

```

Therefore, it is sufficient to have algorithms of computing all subsumee projection justifications for query languages ran and Ran .

The axioms of the form $\text{ran}(r) \sqsubseteq X$ might cause non-trivial entailments. For example, let $\mathcal{T}_1 = \{X \equiv \exists r.Y, Y \equiv A_1 \sqcap A_2\}$, $\mathcal{T}_2 = \mathcal{T}_1 \cup \{\text{ran}(r) \sqsubseteq A_1\}$, $\Sigma = \{X, A_1, A_2, r\}$. Then we have $\mathcal{T}_1 \models \exists r.(A_1 \sqcap A_2)$. However, as the axiom $\alpha \text{ran}(r) \sqsubseteq A_1 \in \mathcal{T}_2$, the conjunct A_1 of Y is already covered by α . Hence, when we define the notion of Σ -entailment, additional parameter $\zeta \in \{\varepsilon\} \cup (N_R \cap \Sigma)$ is used. We call this additional parameter *context of a role*, i.e. an expression of the form $\text{ran}(\zeta)$. We treat ε as a special role name and set $\text{ran}(\varepsilon) = \top$. The set of all role contexts, in symbols \mathcal{C}^{Σ} , is defined as $\mathcal{C}^{\Sigma} = \{\varepsilon\} \cup (N_R \cap \Sigma)$.

For a signature Σ , let $\Sigma^{\text{dom}} = \{\text{dom}(t) \mid t \in N_R \cap \Sigma\}$ and $\Sigma^{\text{ran}} = \{\text{ran}(t) \mid t \in N_R \cap \Sigma\}$ be the sets consisting of concepts of the form $\text{dom}(t)$ and $\text{ran}(t)$ for every role name t in Σ , respectively. Furthermore let $\Sigma^{(\text{ran}, \zeta)} = \Sigma \cup \Sigma^{\text{dom}} \cup \{\text{ran}(\zeta) \mid \zeta \neq \varepsilon\}$ and $\Sigma^{(\text{Ran}, \zeta)} = \Sigma \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}$, for $\zeta \in \mathcal{C}^{\Sigma}$. Note that $\Sigma^{(\text{Ran}, \zeta)} = \Sigma^{(\text{Ran}, \zeta')}$ for every $\zeta, \zeta' \in \mathcal{C}^{\Sigma}$. For example, let $\Sigma = \{A, B, r\}$ be a signature. Then $\Sigma^{\text{dom}} = \{\text{dom}(r)\}$, $\Sigma^{\text{ran}} = \{\text{ran}(s)\}$, $\mathcal{C}^{\Sigma} = \{r, \varepsilon\}$, $\Sigma^{(\text{ran}, \zeta)} = \{A, B, r, \text{dom}(r), \text{ran}(\zeta)\}$ and $\Sigma^{(\text{Ran}, \zeta)} = \{A, B, r, \text{dom}(r), \text{ran}(r)\}$.

In order to deal with Ran -subsumption query, we extend Definition 86 in Section 3.2.2 and get the following definitions.

Algorithm 22: Computing the set of all Subsumer Projection Justifications

```

1 function COVER $\rightarrow$  ( $\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L}$ )
2   if  $\mathcal{L} \in \{\text{ran}, \text{Ran}\}$  then
3      $\mathbb{M}_{(X_1, X_2)}^{\rightarrow} := \text{COVER}_{\rightarrow}^{\text{Nc}}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2) \otimes \text{COVER}_{\rightarrow}^{1, \exists}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{Ran})$ 
4   end
5   if  $\mathcal{L} = \text{Ran}^{\square, u}$  then
6      $\mathbb{M}_{(X_1, X_2)}^{\rightarrow} := \{\emptyset\}$ 
7     for every  $\psi_1 \in F_{\mathcal{T}_1}(X_1)$  do
8        $\mathbb{M}_{\psi_1}^{\rightarrow} := \{\emptyset\}$ 
9       for every  $\psi_2 \in F_{\mathcal{T}_2}(X_2)$  do
10         $\mathbb{M}_{\psi_1, \psi_2}^{\rightarrow} :=$ 
11           $\text{COVER}_{\rightarrow}^{\text{Nc}}(\mathcal{T}_1, \psi_1, \Sigma, \mathcal{T}_2, \psi_2) \otimes \text{COVER}_{\rightarrow}^{2, \exists}(\mathcal{T}_1, \psi_1, \Sigma, \mathcal{T}_2, \psi_2, \text{Ran}^{\square, u})$ 
12         $\mathbb{M}_{\psi_1}^{\rightarrow} := \mathbb{M}_{\psi_1}^{\rightarrow} \cup \mathbb{M}_{\psi_1, \psi_2}^{\rightarrow}$ 
13      end
14     $\mathbb{M}_{(X_1, X_2)}^{\rightarrow} = \mathbb{M}_{(X_1, X_2)}^{\rightarrow} \otimes \mathbb{M}_{\psi_1}^{\rightarrow}$ 
15  end
16  return  $\text{Minimise}_{\subseteq}(\mathbb{M}_{(X_1, X_2)}^{\rightarrow})$ 

```

Definition 158 (Σ -Entailment) Let \mathcal{T} be a normalised \mathcal{ELH}^r -terminology and Σ be a signature. For $\mathcal{L} \in \{\text{ran}, \text{Ran}\}$, $\zeta \in \mathbb{N}_R \cup \{\varepsilon\}$, $A \in \mathbb{N}_C$ and $s \in \mathbb{N}_R$, the following statements hold:

- (i) $A \in \mathbb{N}_C$ is $(\Sigma, \text{ran}, \zeta)$ -entailed in \mathcal{T} iff there is an \mathcal{EL}_{Σ} -concept C such that $\mathcal{T} \models \text{ran}(\zeta) \sqcap C \sqsubseteq A$;
- (ii) $A \in \mathbb{N}_C$ is $(\Sigma, \text{Ran}, \zeta)$ -entailed in \mathcal{T} iff there is an $\mathcal{EL}_{\Sigma}^{\text{ran}}$ -concept D such that $\mathcal{T} \models D \sqsubseteq A$;
- (iii) $s \in \mathbb{N}_R$ is Σ -entailed in \mathcal{T} iff there exists $s' \in \mathbb{N}_R \cap \Sigma$ such that $\mathcal{T} \models s' \sqsubseteq s$.

We then get the following definition by extending the Definition 87 in Section 3.2.2 for Ran-subsumption query.

Definition 159 (Complex Σ -Entailment) Let \mathcal{T} be a normalised and acyclic \mathcal{ELH}^r -terminology, let Σ be a signature, and let $X \in \mathbb{N}_C$.

We say that X is complex $\langle \Sigma, \mathcal{L}, \zeta \rangle$ -entailed w.r.t. \mathcal{T} iff for every $Y \in \text{non-conj}_{\mathcal{T}}(X)$ one of the following conditions holds:

1. there exists $B \in \Sigma^{(\mathcal{L}, \zeta)}$ such that $\mathcal{T} \models B \sqsubseteq Y$ and $\mathcal{T} \not\models B \sqsubseteq X$;
2. there exists $Y \equiv \exists r.Z \in \mathcal{T}$ and r and Z are each $\langle \Sigma, \mathcal{L}, \zeta \rangle$ -entailed in \mathcal{T} .

Otherwise, X is said to be simply $\langle \Sigma, \mathcal{L}, \zeta \rangle$ -entailed.

The concept name X being complex Sigma-entailed w.r.t. \mathcal{T} states that every conjunct of X is either defined as a Σ -entailed existential restriction (Condition 2) or it is entailed by a Σ -concept name, dom- or ran-concept by which X is not entailed.

We now define the notion of a *subsumee simulation* from \mathcal{T}_1 to \mathcal{T}_2 in a role context as a subset of $\text{sig}^{\text{Nc}}(\mathcal{T}_1) \times \text{sig}^{\text{Nc}}(\mathcal{T}_2) \times \mathcal{C}_{\mathcal{T}_1}^{\Sigma}$, where $\mathcal{C}_{\mathcal{T}_1}^{\Sigma} := \{\varepsilon\} \cup (\mathbb{N}_{\mathbb{R}} \cap (\Sigma \cup \text{sig}(\mathcal{T}_1)))$ is the range of role contexts.

We extend Definition 88 and obtain the following notion of subsumee simulation. The new definition works also for Ran-subsumption queries.

Definition 160 (Subsumee Simulation) *Let \mathcal{T}_1 and \mathcal{T}_2 be two normalised $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies, and let Σ be a signature. A relation $S \subseteq \text{sig}^{\text{Nc}}(\mathcal{T}_1) \times \text{sig}^{\text{Nc}}(\mathcal{T}_2) \times \mathcal{C}_{\mathcal{T}_1}^{\Sigma}$ is a $\langle \Sigma, \mathcal{L} \rangle$ -subsumee simulation from \mathcal{T}_1 to \mathcal{T}_2 iff S satisfies the following conditions:*

- $(S_{\text{Nc}}^{\leftarrow, \mathcal{L}})$ if $(X_1, X_2, \zeta) \in S$, then for every $\varphi \in \Sigma^{\langle \mathcal{L}, \zeta \rangle}$ and for every $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$ with $\mathcal{T}_2 \not\models \text{ran}(\zeta) \sqsubseteq X'_2$, $\mathcal{T}_1 \models \varphi \sqsubseteq X_1$ implies $\mathcal{T}_2 \models \varphi \sqsubseteq X'_2$;
- $(S_{\exists}^{\leftarrow, \mathcal{L}})$ if $(X_1, X_2, \zeta) \in S$ and $X_1 \equiv \exists r.Y_1 \in \mathcal{T}_1$ such that $\mathcal{T}_1 \models s \sqsubseteq r$ for $s \in \Sigma$ and Y_1 is (Σ, \mathcal{L}, s) -entailed in \mathcal{T}_1 , then for every $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$ not entailed by $\text{dom}(s)$ or $\text{ran}(\zeta)$ w.r.t. \mathcal{T}_2 , there exists $X'_2 \equiv \exists r'.Y_2 \in \mathcal{T}_2$ such that $\mathcal{T}_2 \models s \sqsubseteq r'$ and $(Y_1, Y_2, s) \in S$;
- $(S_{\sqcap}^{\leftarrow, \mathcal{L}})$ if $(X_1, X_2, \zeta) \in S$ and $X_1 \equiv Y_1 \sqcap \dots \sqcap Y_n \in \mathcal{T}_1$, then for every $Y_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$ not entailed by $\text{ran}(\zeta)$ w.r.t. \mathcal{T}_2 , there exists $Y_1 \in \text{non-conj}_{\mathcal{T}_1}(X_1)$ not entailed by $\text{ran}(\zeta)$ w.r.t. \mathcal{T}_2 with $(Y_1, Y_2, \varepsilon) \in S$.

We write $\mathcal{T}_1 \sim_{\Sigma, \text{ran}}^{\leftarrow} \mathcal{T}_2$ iff there is a Σ -subsumer simulation S from \mathcal{T}_1 to \mathcal{T}_2 such that for every $A, r \in \Sigma$: $(A, A, \varepsilon) \in S$ and $(A, A, r) \in S$.

We write $\mathcal{T}_1 \sim_{\Sigma, \text{Ran}}^{\leftarrow} \mathcal{T}_2$ iff there is a Σ -subsumer simulation S from \mathcal{T}_1 to \mathcal{T}_2 such that for every $A \in \mathbb{N}_{\mathbb{C}} \cap \Sigma$: $(A, A, \varepsilon) \in S$.

For $\mathcal{L} \in \{\text{ran}, \text{Ran}\}$ and $\zeta \in \Sigma \cap \mathbb{N}_{\mathbb{R}}$, we write $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \mathcal{L}} \langle \mathcal{T}_2, X_2 \rangle$ iff there is a Σ -subsumer simulation S from \mathcal{T}_1 to \mathcal{T}_2 with $(X_1, X_2, \zeta) \in S$ for which $\mathcal{T}_1 \sim_{\Sigma, \mathcal{L}}^{\leftarrow} \mathcal{T}_2$.

Analogously to subsumer simulations, a subsumee simulation captures the set of subsumees as it is made precise in the following theorem from [34].

Theorem 161 *We have the following conclusions:*

- $\mathcal{T}_1 \sim_{\Sigma, \text{ran}}^{\leftarrow} \mathcal{T}_2$ iff $\text{rhsWtn}_{\Sigma}^{\text{ran}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$;
- $\mathcal{T}_1 \sim_{\Sigma, \text{Ran}}^{\leftarrow} \mathcal{T}_2$ iff $\text{rhsWtn}_{\Sigma}^{\text{Ran}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$.

Theorem 162 *Let $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \mathcal{L}} \langle \mathcal{T}_2, X_2 \rangle$. Then:*

- $Q = \text{ran}$: for every $C \in \mathcal{E}\mathcal{L}_{\Sigma}$ with $\mathcal{T}_1 \models \text{ran}(\zeta) \sqcap C \sqsubseteq X_1$, it holds that $\mathcal{T}_2 \models \text{ran}(\zeta) \sqcap C \sqsubseteq X_2$;

Algorithm 23: Computing the set of all Subsumee Projection Justifications ($S_{N_c}^{\leftarrow, \mathcal{L}}$)

```

1 function COVER $_{\leftarrow}^{N_c}$  ( $\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L}, \zeta$ )
2    $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} = \{\emptyset\}$ 
3   for every  $B \in \Sigma^{(\mathcal{L}, \zeta)}$  such that  $\mathcal{T}_1 \models B \sqsubseteq X_1$  do
4     for every  $X_2 \in \text{non-conj}_{\mathcal{T}_2}(X_1)$  such that  $\zeta = \varepsilon$  or  $\mathcal{T}_2 \models \text{ran}(\zeta) \sqsubseteq X_2$  do
5        $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \mathbb{M}_{(X_1, X_2)}^{\leftarrow} \otimes \text{Just}_{\mathcal{T}_2}(B \sqsubseteq X_2)$ 
6     end
7   end
8   return  $\text{Minimise}_{\subseteq}(\mathbb{M}_{(X_1, X_2)}^{\leftarrow})$ 

```

- $Q = \text{Ran}$: for every $C \in \mathcal{E} \mathcal{L}_{\Sigma}^{\text{ran}}$ with $\mathcal{T}_1 \models C \sqsubseteq X_1$, it holds that $\mathcal{T}_2 \models C \sqsubseteq X_2$.

In order to collect the axioms necessary to satisfy the conditions of Definition 160, we develop Algorithm 23 (for Case ($S_{N_c}^{\leftarrow, \mathcal{L}}$)), Algorithm 24 (for Case ($S_{\exists}^{\leftarrow, \mathcal{L}}$)) and Algorithm 25 (for Case ($S_{\sqcap}^{\leftarrow, \mathcal{L}}$)).

The algorithm for computing subsumee projection justifications relies on the notion of subsumee simulation between terminologies [12, 34]. First we present some auxiliary notions for handling conjunctions on the left-hand side of subsumptions.

Like in Section 3.1.3, we define for each concept name X a so-called *definitorial forest* consisting of sets of axioms of the form $Y \equiv Y_1 \sqcap \dots \sqcap Y_n$ which can be thought of as forming *trees*. Any $\langle X, \Sigma \rangle$ -subsumee projection justification contains the axioms of a selection of these trees, i.e., one tree for every conjunction formulated over Σ that entails X w.r.t. \mathcal{T} .

In our algorithms, definitorial forest is used to enumerate all possible situations and find all possible sub-trees such that $\text{Case}(S_{\exists}^{\leftarrow, \mathcal{L}})$ can be satisfied.

For the next algorithm, we recall the definition of $\text{def}_{\mathcal{T}}^{\sqcap}$. $\text{def}_{\mathcal{T}}^{\sqcap} := \{X \in \text{sig}^{N_c}(\mathcal{T}) \mid X \equiv Y_1 \sqcap \dots \sqcap Y_n \in \mathcal{T}\}$ is the set of concept names that are conjunctively defined in \mathcal{T} . For every $X \in \text{def}_{\mathcal{T}}^{\sqcap}$, we set $\text{def}_{\mathcal{T}}^{\sqcap}(X) := \alpha$, where $\alpha = X \equiv Y_1 \sqcap \dots \sqcap Y_n \in \mathcal{T}$.

The axiom $\alpha_{X_1} := X_1 \equiv Y_1 \sqcap \dots \sqcap Y_m \in \mathcal{T}_1$ in Line 2 of Algorithm 25 is guaranteed by Line 12 of Algorithm 26. In the case where X_2 is defined as a conjunction in \mathcal{T}_2 , the pair consisting of \mathcal{T}_2 containing only a partial conjunctive tree rooted at X_2 and X_2 needs to be considered to be sufficient to subsumee simulate X_1 in \mathcal{T}_1 . Therefore Algorithm 25 considers every partial conjunctive tree Γ from $\text{DefForest}_{\mathcal{T}_2}^{\sqcap}(X_2)$ in Line 4 and removes the axioms in δ_{Γ} connecting the leaves of Γ with the remaining conjunctive tree from \mathcal{T}_2 in Lines 11.

The following theorem shows the correctness of the algorithms 26 can compute minimal subset of \mathcal{T}_2 that maintains subsumee simulation w.r.t. ran-subsumption query.

Algorithm 24: Computing the set of all Subsumee Projection Justifications ($S_{\exists}^{\leftarrow, \mathcal{L}}$)

```

1 function COVER $_{\exists}^{\leftarrow}$  ( $\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L}, \zeta$ )
2   Let  $\alpha_{X_1} := X_1 \equiv \exists r. Y_1 \in \mathcal{T}_1$ 
3    $M_{(X_1, X_2)}^{\leftarrow} := \{\text{max-tree}_{\mathcal{T}_2}^{\square}(X_2)\}$ 
4   for every  $s \in \Sigma \cap N_R$  such that  $\mathcal{T}_1 \models s \sqsubseteq r$  do
5     for every  $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$  such that  $\zeta \neq \varepsilon$  implies  $\mathcal{T}_2 \not\models \text{ran}(\zeta) \sqsubseteq X'_2$  and
6        $\mathcal{T}_2 \not\models \text{dom}(s) \sqsubseteq X'_2$  do
7         Let  $\alpha_{X'_2} := X'_2 \equiv \exists r'. Y'_2 \in \mathcal{T}_2$ 
8          $M_{Y'_2}^{\leftarrow} := \text{COVER}_{\leftarrow}(\mathcal{T}_1, Y_1, \Sigma, \mathcal{T}_2, Y'_2, \mathcal{L}, s)$ 
9          $M_{(X_1, X_2)}^{\leftarrow} := M_{(X_1, X_2)}^{\leftarrow} \otimes \{\{\alpha_{X'_2}\}\} \otimes \text{Just}_{\mathcal{T}_2}(s \sqsubseteq r) \otimes M_{Y'_2}^{\leftarrow}$ 
10      end
11  end
12  return  $\text{Minimise}_{\subseteq}(M_{(X_1, X_2)}^{\leftarrow})$ 

```

Algorithm 25: Computing the set of all Subsumee Projection Justifications ($S_{\square}^{\leftarrow, \mathcal{L}}$)

```

1 function COVER $_{\square}^{\leftarrow}$  ( $\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L}, \zeta$ )
2   let  $\alpha_{X_1} := X_1 \equiv Y_1 \square \dots \square Y_m \in \mathcal{T}_1$ 
3    $M_{(X_1, X_2)}^{\leftarrow} := \emptyset$ 
4   for every  $\Gamma \in \text{DefForest}_{\mathcal{T}_2}^{\square}(X_2)$  do
5     Let  $\delta_{\Gamma} := \{\text{def}_{\mathcal{T}_2}^{\square}(X') \mid X' \in \text{leaves}(\Gamma) \cap \text{def}_{\mathcal{T}_2}^{\square}\}$ 
6      $M_{\Gamma}^{\leftarrow} := \{\Gamma\}$ 
7     for every  $X'_2 \in \text{leaves}(\Gamma)$  such that  $\zeta = \varepsilon$  or  $\mathcal{T}_2 \not\models \text{ran}(\zeta) \sqsubseteq X'_2$  do
8        $M_{X'_2}^{\leftarrow} := \emptyset$ 
9       for every  $X'_1 \in \text{non-conj}_{\mathcal{T}_1}(X_1)$  such that  $\zeta = \varepsilon$  or  $\mathcal{T}_2 \not\models \text{ran}(\zeta) \sqsubseteq X'_1$  do
10        if  $\langle \mathcal{T}_1, X'_1 \rangle \sim_{\Sigma, \varepsilon}^{\leftarrow, \mathcal{L}} \langle \mathcal{T}_2 \setminus \delta_{\Gamma}, X'_2 \rangle$  then
11           $M_{X'_2}^{\leftarrow} := M_{X'_2}^{\leftarrow} \cup \text{COVER}_{\leftarrow}(\mathcal{T}_1, X'_1, \Sigma, \mathcal{T}_2 \setminus \delta_{\Gamma}, X'_2, \mathcal{L}, \varepsilon)$ 
12        end
13      end
14       $M_{\Gamma}^{\leftarrow} := M_{\Gamma}^{\leftarrow} \otimes M_{X'_2}^{\leftarrow}$ 
15    end
16     $M_{(X_1, X_2)}^{\leftarrow} := M_{(X_1, X_2)}^{\leftarrow} \cup M_{\Gamma}^{\leftarrow}$ 
17  end
18  return  $\text{Minimise}_{\subseteq}(M_{(X_1, X_2)}^{\leftarrow})$ 

```

Algorithm 26: Computing the set of all Subsumee Projection Justifications

```

1 function COVER $\leftarrow$ ( $\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L}, \zeta$ )
2   if  $X_1$  is not  $\langle \Sigma, \mathcal{L}, \zeta \rangle$ -entailed w.r.t.  $\mathcal{T}_1$  then
3     |   return  $\{\emptyset\}$ 
4   end
5    $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \text{COVER}_{\leftarrow}^{\text{Nc}}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L}, \zeta)$ 
6   if  $X_1$  is not complex  $\langle \Sigma, \mathcal{L}, \zeta \rangle$ -entailed-entailed in  $\mathcal{T}_1$  then
7     |   return  $\text{Minimise}_{\subseteq}(\mathbb{M}_{(X_1, X_2)}^{\leftarrow})$ 
8   end
9   if  $X_1 \equiv \exists r.Y \in \mathcal{T}_1$ , and  $r, Y$  are  $\Sigma$ -entailed w.r.t.  $\mathcal{T}_1$  then
10    |    $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \mathbb{M}_{(X_1, X_2)}^{\leftarrow} \otimes \text{COVER}_{\leftarrow}^{\exists}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L}, \zeta)$ 
11    end
12    else if  $X_1 \equiv Y_1 \sqcap \dots \sqcap Y_m \in \mathcal{T}_1$  then
13      |    $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \mathbb{M}_{(X_1, X_2)}^{\leftarrow} \otimes \text{COVER}_{\leftarrow}^{\sqcap}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L}, \zeta)$ 
14      end
15    return  $\text{Minimise}_{\subseteq}(\mathbb{M}_{(X_1, X_2)}^{\leftarrow})$ 

```

Theorem 163 Let $\chi = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L} \rangle$, where \mathcal{T}_1 and \mathcal{T}_2 are two acyclic and normalised $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies, Σ is a signature and $X_1, X_2 \in \Sigma \cap \text{Nc}$. Additionally, let

$$\mathbb{M} := \text{COVER}_{\leftarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran}, \zeta).$$

If $\mathcal{T}_1 \sim_{\Sigma}^{\leftarrow} \mathcal{T}_2$, then for every $\mathcal{M} \in \mathbb{M}$, it holds that $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \mathcal{L}} \langle \mathcal{M}, X_2 \rangle$ and \mathcal{M} is minimal w.r.t. set inclusion.

The following theorem shows that the set of all subsumee projection justifications can be computed by Algorithm 26.

Theorem 164 Let $\chi_{\text{ran}} = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran} \rangle$, $\chi_{\text{Ran}} = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{Ran} \rangle$, where \mathcal{T}_1 and \mathcal{T}_2 are two acyclic and normalised $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies, Σ is a signature and $X_1, X_2 \in \Sigma \cap \text{Nc}$. Additionally, let

$$\mathbb{M}_{\text{ran}} := \text{Minimise}_{\subseteq}(\text{COVER}_{\leftarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran}, \varepsilon) \otimes \bigotimes_{r \in \Sigma \cap \text{Nr}} \text{COVER}_{\leftarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran}, r)),$$

$$\mathbb{M}_{\text{Ran}} := \text{COVER}_{\leftarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran}, \varepsilon).$$

If $\mathcal{T}_1 \sim_{\Sigma}^{\leftarrow} \mathcal{T}_2$, then \mathbb{M}_{ran} is the set of all subsumee justifications under χ_{Ran} and \mathbb{M}_{Ran} is the set of all subsumee justifications under χ_{ran} .

Theorem 164 follows from Definition 137, Theorem 162 and Theorem 163.

The number of (minimal) projection justifications depends on \mathcal{T}_1 , \mathcal{T}_2 and Σ and exponential bounded in the size of the terminologies. The simulation checks can be performed in polynomial time [12, 34]. Algorithm 26 runs in exponential time in the number of axioms contained in the input terminologies, in the worst case. On the one hand, the algorithm uses justifications (see Line 6 of Alg. 24 and Line 5 of Alg. 23) whose number grows exponentially for role inclusions as well as concept name inclusions. The different justifications are each incorporated using the operator \otimes resulting in possibly different subsumption justifications. The majority of the running time will be spent on computing justifications. Another source of exponential blowup is contained in Line 4 of Algorithm 25. The number of elements in the set $\text{DefForest}_{\mathcal{T}}^{\square}(X)$ grows exponentially in $|\mathcal{T}|$. According to our experience so far, however, it seems plausible to assume that definitorial forests in practical ontologies remain rather small and, thus, they do not cause a serious slowdown of the algorithm.

6.4 Proofs

6.4.1 Proof Regarding Subsumption Projection Justification

Lemma 152 *Let $\chi(\varphi) = \langle \mathcal{T}_1, \varphi, \Sigma, \mathcal{T}_2, \varphi, \mathcal{L} \rangle$ and $\chi(A) = \langle \mathcal{T}_1, A, \Sigma, \mathcal{T}_2, A, \mathcal{L} \rangle$, where $\varphi \in (\Sigma \cap \text{N}_C) \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}$, $A \in \Sigma \cap \text{N}_C$ and $\mathcal{L} \in \{\text{ran}, \text{Ran}, \text{Ran}^{\square, u}\}$. Then we have that for every $\mathcal{J}_{\chi(\varphi)} \in \mathbb{J}_{\chi(\varphi)}^{\rightarrow}$, $\varphi \notin \text{lhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{J}_{\chi(\varphi)})$, and for every $\mathcal{J}_{\chi(A)} \in \mathbb{J}_{\chi(A)}^{\leftarrow}$, $A \notin \text{rhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{J}_{\chi(A)})$.*

Proof We first prove that $\varphi \notin \text{lhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{J}_{\chi(\varphi)})$, where $\mathcal{L} = \{\text{ran}, \text{Ran}, \text{Ran}^{\square, u}\}$ for every $\mathcal{J}_{\chi(\varphi)} \in \mathbb{J}_{\chi(\varphi)}^{\rightarrow}$, $\varphi \notin \text{lhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{J}_{\chi(\varphi)})$. According to the Theorem 40 and 61 in [23], we have that the sets $\text{lhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{J}_{\chi(\varphi)}) \subseteq (\Sigma \cap \text{N}_C) \cup \{\text{dom}(r) \mid r \in \Sigma\} \cup \{\text{ran}(r) \mid r \in \Sigma\}$ of *left-hand* subsumption query difference witnesses consist of the left-hand sides of the type- δ_4 inclusions in $\text{cDiff}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2)$. As $\delta_4 ::= A \sqsubseteq C \mid \text{dom}(r) \sqsubseteq C \mid \text{ran}(r) \sqsubseteq C$, we can get that $\varphi \notin \text{lhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{J}_{\chi(\varphi)})$ for every $\mathcal{J}_{\chi(\varphi)} \in \mathbb{J}_{\chi(\varphi)}^{\rightarrow}$ by Definition 137.

Then we prove that $A \notin \text{rhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, J_A)$ for every $J_A \in \mathcal{J}_{\chi(A)}^{\leftarrow}$. According to the Theorem 40 and 61 in [23], we have that the sets $\text{rhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2) \subseteq \text{N}_C \cap \Sigma$ of *right-hand* subsumption query difference witnesses consist of the right-hand sides of type- δ_2 and type- δ_3 inclusions in $\text{cDiff}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, \mathcal{T}_2)$, respectively, depending on the query language \mathcal{L} . We distinguish according to the different query languages:

- when $\mathcal{L} = \text{Ran}$ or $\text{Ran}^{\square, u}$: we consider type- δ_3 inclusions. As $\delta_3 ::= D \sqsubseteq A$ and Definition 137, we have that $A \notin \text{rhsWtn}_{\Sigma}^{\mathcal{L}}(\mathcal{T}_1, J_A)$ for every $J_A \in \mathcal{J}_{\chi(A)}^{\leftarrow}$.
- when $\mathcal{L} = \text{ran}$: we consider type- δ_2 inclusions. As $\delta_2 ::= C \sqsubseteq A \mid \text{ran}(r) \sqcap C \sqsubseteq A$, we have that $A \notin \text{rhsWtn}_{\Sigma}^{\text{ran}}(\mathcal{T}_1, J_A)$ for every $J_A \in \mathcal{J}_{\chi(A)}^{\leftarrow}$.

□

Theorem 153 Let $\mathcal{L} \in \{\text{ran}, \text{Ran}, \text{Ran}^{\square, u}\}$ and $\mathbb{M}_\rho^\mathcal{L}$ be the set of all \mathcal{L} -projection modules under $\rho = \langle \mathcal{T}_1, \Sigma, \mathcal{T}_2 \rangle$ that are minimal w.r.t. \sqsubset . Then

$$\mathbb{M}_\rho^\mathcal{L} = \text{Minimize}_{\sqsubset} \left(\mathbb{J}_\rho^R \otimes \bigotimes_{\varphi \in (\Sigma \cap \mathbb{N}_C) \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}} \mathbb{J}_{\chi(\varphi)}^{\rightarrow} \otimes \bigotimes_{A \in \Sigma \cap \mathbb{N}_C} \mathbb{J}_{\chi(A)}^{\leftarrow} \right)$$

where $\chi(\psi) = \langle \mathcal{T}_1, \psi, \Sigma, \mathcal{T}_2, \psi, \mathcal{L} \rangle$.

Proof Let \mathbb{S}_1 and \mathbb{S}_2 be sets of sets that are each minimal w.r.t. \sqsubset , i.e., they satisfy: $\forall S \in \mathbb{S}_i$ with $i \in \{1, 2\}$, there is no $S' \in \mathbb{S}_i$ such that $S \subset S'$. We say that $\mathbb{S}_1 \stackrel{*}{\subseteq} \mathbb{S}_2$ iff for every $S_2 \in \mathbb{S}_2$, there exists an $S_1 \in \mathbb{S}_1$ such that $S_1 \subseteq S_2$.

Then let

$$\mathbb{S}_\rho^\mathcal{L} = \text{Minimize}_{\sqsubset} \left(\mathbb{J}_\rho^R \otimes \bigotimes_{\varphi \in (\Sigma \cap \mathbb{N}_C) \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}} \mathbb{J}_{\chi(\varphi)}^{\rightarrow} \otimes \bigotimes_{A \in \Sigma \cap \mathbb{N}_C} \mathbb{J}_{\chi(A)}^{\leftarrow} \right)$$

It can readily be seen that $\mathbb{S}_1 \stackrel{*}{\subseteq} \mathbb{S}_2$ and $\mathbb{S}_2 \stackrel{*}{\subseteq} \mathbb{S}_1$ implies $\mathbb{S}_1 = \mathbb{S}_2$, using the property of minimality w.r.t. \sqsubset of \mathbb{S}_1 and \mathbb{S}_2 . Therefore, to show the lemma, it is sufficient to show that $\mathbb{M}_\rho^\mathcal{L} \stackrel{*}{\subseteq} \mathbb{S}_\rho^\mathcal{L}$ and $\mathbb{S}_\rho^\mathcal{L} \stackrel{*}{\subseteq} \mathbb{M}_\rho^\mathcal{L}$.

We first show that $\mathbb{M}_\rho^\mathcal{L} \stackrel{*}{\subseteq} \mathbb{S}_\rho^\mathcal{L}$. By Theorem 24, the set of all \mathcal{L} -subsumption difference witnesses is defined as

$$\text{Wtn}_\Sigma^\mathcal{L}(\mathcal{T}_1, \mathcal{T}_2) = \text{roleWtn}_\Sigma^\mathcal{L}(\mathcal{T}_1, \mathcal{T}_2) \cup \text{lhsWtn}_\Sigma^\mathcal{L}(\mathcal{T}_1, \mathcal{T}_2) \cup \text{rhsWtn}_\Sigma^\mathcal{L}(\mathcal{T}_1, \mathcal{T}_2),$$

where the set $\text{roleWtn}_\Sigma^\mathcal{L}(\mathcal{T}_1, \mathcal{T}_2) = \{r \in \Sigma \cap \mathbb{N}_R \mid r \sqsubseteq s \in \text{cDiff}_\Sigma^\mathcal{L}(\mathcal{T}_1, \mathcal{T}_2) \text{ or } s \sqsubseteq r \in \text{cDiff}_\Sigma^\mathcal{L}(\mathcal{T}_1, \mathcal{T}_2)\}$, the set $\text{lhsWtn}_\Sigma^\mathcal{L}(\mathcal{T}_1, \mathcal{T}_2) = \{\varphi \in (\Sigma \cap \mathbb{N}_C) \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}} \mid \varphi \sqsubseteq \text{rhs}(\alpha) \text{ and } \alpha \text{ is a type-}\delta_4 \text{ or type-}\delta_5 \text{ inclusions in the set } \text{cDiff}_\Sigma^\mathcal{L}(\mathcal{T}_1, \mathcal{T}_2)\}$ and the set $\text{rhsWtn}_\Sigma^\mathcal{L}(\mathcal{T}_1, \mathcal{T}_2) = \{A \in \Sigma \cap \mathbb{N}_C \mid \text{lhs}(\alpha) \sqsubseteq A \text{ and } \alpha \text{ is a type-}\delta_2 \text{ or type-}\delta_3 \text{ inclusions in the set } \text{cDiff}_\Sigma^\mathcal{L}(\mathcal{T}_1, \mathcal{T}_2)\}$ respectively, depending on the query language \mathcal{L} . With Lemma 148, Lemma 152 and Lemma 72, we have that $\text{Wtn}_\Sigma^\mathcal{L}(\mathcal{T}_1, M) = \emptyset$ for every $M \in \mathbb{S}_\rho^\mathcal{L}$. Consequently, by Definition 137 and Definition 19, every $M \in \mathbb{S}_\rho^\mathcal{L}$ is a projection module under ρ , which means there exists $M' \in \mathbb{M}_\rho^\mathcal{L}$ such that $M' \subseteq M$. Hence, $\mathbb{M}_\rho^\mathcal{L} \stackrel{*}{\subseteq} \mathbb{S}_\rho^\mathcal{L}$.

Now we show that $\mathbb{S}_\rho^\mathcal{L} \stackrel{*}{\subseteq} \mathbb{M}_\rho^\mathcal{L}$. Let $M \in \mathbb{M}_\rho^\mathcal{L}$ be a minimal projection module under ρ . Since $\text{Wtn}_\Sigma^\mathcal{L}(\mathcal{T}_1, M) = \emptyset$, there exists $\mathcal{J}_R \in \mathbb{J}_\rho^R$ such that $\mathcal{J}_R \subseteq M$; for every $\varphi \in (\Sigma \cap \mathbb{N}_C) \cup \Sigma^{\text{dom}} \cup \Sigma^{\text{ran}}$, there exists $\mathcal{J}_\varphi \in \mathbb{J}_{\chi(\varphi)}^{\rightarrow}$ such that $\mathcal{J}_\varphi \subseteq M$; and for every $A \in \Sigma \cap \mathbb{N}_C$, there exists $\mathcal{J}_A \in \mathbb{J}_{\chi(A)}^{\leftarrow}$ such that $\mathcal{J}_A \subseteq M$. Consequently, there exists $S \in \mathbb{S}_\rho^\mathcal{L}$ such that $S \subseteq M$. Hence, $\mathbb{S}_\rho^\mathcal{L} \stackrel{*}{\subseteq} \mathbb{M}_\rho^\mathcal{L}$. \square

6.4.2 Canonical Model & Helpful Lemmas

The material of this section comes from an unpublished appendix of [34]. As it has not been published yet, we list relevant definitions and lemmas in this section in order to make the proofs in this thesis

self-contained. We assume that N_I is countably infinite set of *individual names* disjoint with N_C and N_R . An *ABox* \mathcal{A} is a non-empty finite set of assertions of the form $\top(a)$, $A(a)$ and $r(a,b)$, where $a,b \in N_I$, $A \in N_C$, and $r \in N_R$. By $\text{obj}(\mathcal{A})$ we denote the set of individual names in \mathcal{A} . An \mathcal{ELH}_\perp -knowledge base \mathcal{K} (KB) is a pair $(\mathcal{T}, \mathcal{A})$ consisting of an \mathcal{ELH}_\perp -TBox \mathcal{T} and an ABox \mathcal{A} . Assertions of the form $C(a)$ and $r(a,b)$, where $a,b \in N_I$, C an \mathcal{EL} -concept or an $\mathcal{EL}^{\sqcap,u}$ -concept, and $r \in N_R$, are called *instance assertions*.

We define a canonical model, $\mathcal{I}_{\mathcal{K}}$, for \mathcal{ELH}_\perp -knowledge bases \mathcal{K} . $\mathcal{I}_{\mathcal{K}}$ can be constructed in polynomial time and gives the same answers to instance queries as \mathcal{K} ; i.e., $\mathcal{I}_{\mathcal{K}} \models \alpha$ if, and only if, $\mathcal{K} \models \alpha$, for any instance assertion α .

Let $\text{sub}(\mathcal{T})$ denote the set of all subconcepts of concepts used in \mathcal{T} , $\text{sig}^{N_R}(\mathcal{T})$ the set of all role names occurring in \mathcal{T} . Take fresh individual names $x_{\text{ran}(r),D}$ for every $r \in \text{rol}(\mathcal{T})$ and $D \in \text{sub}(\mathcal{T})$ and set

$$N_{\text{aux}} := \{x_{\text{ran}(r),D} \mid r \in \text{sig}^{N_R}(\mathcal{T}) \text{ and } D \in \text{sub}(\mathcal{T})\}.$$

Now define the *generating interpretation* $\mathcal{W}_{\mathcal{K}}$ of a KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ as follows:

$$\begin{aligned} \Delta^{\mathcal{W}_{\mathcal{K}}} &:= \text{obj}(\mathcal{A}) \cup N_{\text{aux}}; \\ A^{\mathcal{W}_{\mathcal{K}}} &:= \{a \in \text{obj}(\mathcal{A}) \mid \mathcal{K} \models A(a)\} \cup \{x_{\text{ran}(r),D} \in N_{\text{aux}} \mid \mathcal{T} \models \text{ran}(r) \sqcap D \sqsubseteq A\}; \\ r^{\mathcal{W}_{\mathcal{K}}} &:= \{(a,b) \in \text{obj}(\mathcal{A}) \times \text{obj}(\mathcal{A}) \mid s(a,b) \in \mathcal{A} \text{ and } \mathcal{T} \models s \sqsubseteq r\} \cup \\ &\quad \{(a, x_{\text{ran}(s),D}) \in \text{obj}(\mathcal{A}) \times N_{\text{aux}} \mid \mathcal{K} \models \exists s.D(a) \text{ and } \mathcal{T} \models s \sqsubseteq r\} \cup \\ &\quad \{(x_{\text{ran}(s),D}, x_{\text{ran}(s'),D'}) \in N_{\text{aux}} \times N_{\text{aux}} \mid \mathcal{T} \models \text{ran}(s) \sqcap D \sqsubseteq \exists s'.D', \mathcal{T} \models s' \sqsubseteq r\}; \\ a^{\mathcal{W}_{\mathcal{K}}} &:= a, \text{ for all } a \in \text{obj}(\mathcal{A}). \end{aligned}$$

A *path* in $\mathcal{W}_{\mathcal{K}}$ is a finite sequence $d_0 r_1 d_1 \cdots r_n d_n$, $n \geq 0$, where $d_0 \in \text{obj}(\mathcal{A})$ and, for all $i < n$, $(d_i, d_{i+1}) \in r_{i+1}^{\mathcal{W}_{\mathcal{K}}}$. We use $\text{paths}(\mathcal{W}_{\mathcal{K}})$ to denote the set of all paths in $\mathcal{W}_{\mathcal{K}}$. If $p \in \text{paths}(\mathcal{W}_{\mathcal{K}})$, then $\text{tail}(p)$ denotes the last element d_n in p .

The *canonical model* $\mathcal{I}_{\mathcal{K}}$ of a knowledge base \mathcal{K} is the restriction of $\mathcal{W}_{\mathcal{K}}$ to all domain elements d such that there is a path in $\mathcal{W}_{\mathcal{K}}$ with tail d . The following result summarizes the main properties of $\mathcal{I}_{\mathcal{K}}$.

Theorem 165 *Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be an \mathcal{ELH}_\perp -KB. Then*

- (i) $\mathcal{I}_{\mathcal{K}}$ is a model of \mathcal{K} ;
- (ii) $\mathcal{I}_{\mathcal{K}}$ can be computed in polynomial time in the size of \mathcal{K} ;
- (iii) for all $x_{C,D} \in \Delta^{\mathcal{I}_{\mathcal{K}}}$ and all $a \in \text{obj}(\mathcal{A})$, if C_0 is a $\mathcal{EL}^{\sqcap,u}$ -concept or of the form $\text{ran}(r)$, then the following holds:
 - $\mathcal{K} \models C_0(a)$ iff $a^{\mathcal{I}_{\mathcal{K}}} \in C_0^{\mathcal{I}_{\mathcal{K}}}$; and
 - $\mathcal{T} \models C \sqcap D \sqsubseteq C_0$ iff $x_{C,D} \in C_0^{\mathcal{I}_{\mathcal{K}}}$.

Proof See Theorem 2 (Extended Version) in [23]. \square

Lemma 166 *Let \mathcal{T} be an $\mathcal{EL}\mathcal{H}^r$ -terminology, let $A \in \mathbf{N}_C$, $r \in \mathbf{N}_R$, and let C be an $\mathcal{EL}^{\square,u}$ -concept. Then the following statements hold:*

- (i) $\mathcal{T} \models A \sqsubseteq C$ iff $(\mathcal{T}, \{A(a)\}) \models C(a)$;
- (ii) $\mathcal{T} \models \text{ran}(r) \sqcap A \sqsubseteq C$ iff $(\mathcal{T}, \{r(a,b), A(b)\}) \models C(b)$;
- (iii) $\mathcal{T} \models \exists r.A \sqsubseteq C$ iff $(\mathcal{T}, \{r(a,b), A(b)\}) \models C(a)$;
- (iv) $\mathcal{T} \models \text{ran}(r) \sqsubseteq C$ iff $(\mathcal{T}, \{r(a,b)\}) \models C(b)$;
- (v) $\mathcal{T} \models \text{dom}(r) \sqsubseteq C$ iff $(\mathcal{T}, \{r(a,b)\}) \models C(a)$.

Proof Follows immediately from the definition of logical entailment w.r.t. TBoxes or knowledge bases. \square

Lemma 167 *Let \mathcal{T} be a normalised $\mathcal{EL}\mathcal{H}^r$ -terminology. Assume $\mathcal{T} \models D \sqsubseteq \exists u.C$, where D is either a concept name or of the form $\text{ran}(r)$, $\text{dom}(r)$, or $\text{ran}(r) \sqcap A$ with $A, r \in \text{sig}(\mathcal{T}) \cup \Sigma$. Then one of the following statements holds:*

- if $\mathcal{T} \models A \sqsubseteq \exists u.C$, then there exist $n \geq 0$ and $r_1, \dots, r_n \in \mathbf{N}_R$ such that $\mathcal{T} \models A \sqsubseteq \exists r_1 \dots \exists r_n.C$;
- if $\mathcal{T} \models \text{ran}(r) \sqcap A \sqsubseteq \exists u.C$, then there exist $n \geq 0$ and $r_1, \dots, r_n \in \mathbf{N}_R$ such that
 - $\mathcal{T} \models \text{ran}(r) \sqcap A \sqsubseteq \exists r_1 \dots \exists r_n.C$, or
 - $\mathcal{T} \models \exists r.A \sqsubseteq \exists r_1 \dots \exists r_n.C$;
- if $\mathcal{T} \models \text{ran}(r) \sqsubseteq \exists u.C$, then there exist $n \geq 0$ and $r_1, \dots, r_n \in \mathbf{N}_R$ such that
 - $\mathcal{T} \models \text{dom}(r) \sqsubseteq \exists r_1 \dots \exists r_n.C$, or
 - $\mathcal{T} \models \text{ran}(r) \sqsubseteq \exists r_1 \dots \exists r_n.C$;
- if $\mathcal{T} \models \text{dom}(r) \sqsubseteq \exists u.C$, then there exist $n \geq 0$ and $r_1, \dots, r_n \in \mathbf{N}_R$ such that
 - $\mathcal{T} \models \text{dom}(r) \sqsubseteq \exists r_1 \dots \exists r_n.C$, or
 - $\mathcal{T} \models \text{ran}(r) \sqsubseteq \exists r_1 \dots \exists r_n.C$.

Proof If $\mathcal{T} \models A \sqsubseteq \exists u.C$, it follows from Lemma 166 that $\mathcal{K} \models \exists u.C(a)$ for $\mathcal{K} = (\mathcal{T}, \{A(a)\})$. Hence, by Theorem 165 we have $a^{\mathcal{I}_{\mathcal{K}}} \in (\exists u.C)^{\mathcal{I}_{\mathcal{K}}}$, i.e. there exists $x \in \Delta^{\mathcal{I}_{\mathcal{K}}}$ with $x \in C^{\mathcal{I}_{\mathcal{K}}}$. If $x \in \text{obj}(\mathcal{A})$, we have $\mathcal{T} \models A \sqsubseteq C$ by Theorem 165 and Lemma 166. We can now assume that $x \notin \text{obj}(\mathcal{A})$, i.e. there exists a finite sequence $d_0 r_1 d_1 \cdots r_n d_n$, $n \geq 1$, where $d_0 = a$, $d_n = x$ and, for all $i < n$, $(d_i, d_{i+1}) \in r_{i+1}^{\mathcal{I}_{\mathcal{K}}}$. By definition of $\mathcal{I}_{\mathcal{K}}$ and by Theorem 165 we have $\mathcal{K} \models (\exists r_1 \dots \exists r_n.C)(a)$, and consequently $\mathcal{T} \models A \sqsubseteq \exists r_1 \dots \exists r_n.C$ by Lemma 166.

If $\mathcal{T} \models \text{ran}(r) \sqcap A \sqsubseteq \exists u.C$, it follows from Lemma 166 that $\mathcal{K} \models \exists u.C(a)$ for

$$\mathcal{K} = (\mathcal{T}, \{r(a, b), A(b)\}).$$

Hence, by Theorem 165 we have $a^{\mathcal{I}_{\mathcal{K}}} \in (\exists u.C)^{\mathcal{I}_{\mathcal{K}}}$, i.e. there exists $x \in \Delta^{\mathcal{I}_{\mathcal{K}}}$ with $x \in C^{\mathcal{I}_{\mathcal{K}}}$. If $x \in \text{obj}(\mathcal{A})$, we either have $\mathcal{T} \models \text{ran}(r) \sqcap A \sqsubseteq C$ or $\mathcal{T} \models \exists r.A \sqsubseteq C$ by Theorem 165 and Lemma 166. We can now assume that $x \notin \text{obj}(\mathcal{A})$, i.e. there exists a finite sequence $d_0 r_1 d_1 \cdots r_n d_n$, $n \geq 1$, where $d_0 \in \text{obj}(\mathcal{A})$, $d_n = x$ and, for all $i < n$, $(d_i, d_{i+1}) \in r_{i+1}^{\mathcal{I}_{\mathcal{K}}}$. By definition of $\mathcal{I}_{\mathcal{K}}$ and by Theorem 165 we have $\mathcal{K} \models (\exists r_1 \dots \exists r_n.C)(d_0)$. Consequently, by Lemma 166, $\mathcal{T} \models \text{ran}(r) \sqcap A \sqsubseteq \exists r_1 \dots \exists r_n.C$ if $d_0 = b$ and $\mathcal{T} \models \exists r.A \sqsubseteq \exists r_1 \dots \exists r_n.C$ otherwise.

The cases for $\mathcal{T} \models \text{dom}(r) \sqsubseteq \exists u.C$ and $\mathcal{T} \models \text{ran}(r) \sqsubseteq \exists u.C$ can be proved analogously to the previous case. \square

In the following we also make use of the following easy equivalence, which allows to characterise conjunctions of roles in terms of simple roles.

Lemma 168 (See [23]) *Let \mathcal{T} be a normalised \mathcal{ELH}^r -terminology, let C be an $\mathcal{EL}^{\text{ran}}$ -concept, and let D be an \mathcal{EL}^{\sqcap} -concept which contains the occurrences $S_i = r_{i,1} \sqcap \dots \sqcap r_{i,m_i}$ of intersections of roles, $1 \leq i \leq k$. Then, $\mathcal{T} \models C \sqsubseteq D$ holds if, and only if, there exist role names s_i , $1 \leq i \leq k$, such that $s_i \sqsubseteq_{\mathcal{T}} r_{i,j}$ for $1 \leq i \leq k$, $1 \leq j \leq m_i$ and $\mathcal{T} \models C \sqsubseteq D'$, where D' is obtained from D by replacing S_i with s_i .*

Lemma 169 *Let \mathcal{T} a normalised \mathcal{ELH}^r -terminology \mathcal{T} and Σ a signature. Additionally, let $\mathcal{V} := \text{N}_C \cup \{ \text{dom}(r), \text{ran}(r) \mid r \in \Sigma \cup \text{sig}(\mathcal{T}) \}$ and $X \in \mathcal{V}$. Let E be an \mathcal{EL}^{\sqcap} -concept such that $\mathcal{T} \models X \sqsubseteq \exists r.D$. Then there exists $Y \bowtie \exists s.Z \in \mathcal{T}$ ($\bowtie \in \{ \sqsubseteq, \equiv \}$) such that $\mathcal{T} \models X \sqsubseteq Y$, $\mathcal{T} \models s \sqsubseteq r$ and $\mathcal{T} \models Z \sqsubseteq D$.*

Proof Let C be the \mathcal{EL} -concept obtained from E by replacing the intersection of roles with simple roles as in Lemma 168. If X is either a concept name or of the form $\text{ran}(r)$ or $\text{dom}(r)$, the lemma can be proved by analysing the structure of the derivation $X \vdash_{\mathcal{T}} \exists r.C$. For the case where $X = \text{ran}(r) \sqcap A$, it follows from Lemma 35 that either $\mathcal{T} \models A \sqsubseteq \exists r.C$ or $\mathcal{T} \models \text{ran}(r) \sqsubseteq \exists r.C$, i.e. we can apply one of the previous cases. \square

6.4.3 Proof Regarding Subsumer Projection Justification

The followings notions are required to handle concepts of the form $\exists u.C$.

Definition 170 Let \mathcal{T} be a normalised $\mathcal{EL}\mathcal{H}^r$ -terminology and a signature Σ . Let $\mathcal{V} := \mathbb{N}_C \cup \{\text{dom}(r), \text{ran}(r) \mid r \in \Sigma \cup \text{sig}(\mathcal{T})\}$ and $\psi \in \mathcal{V}$. The relation $\rightsquigarrow_{\mathcal{T}} \subseteq \mathcal{V} \times \mathcal{V}$ is inductively defined as follows:

- (i) $\{\psi\} \rightsquigarrow_{\mathcal{T}} \{\psi\}$;
- (ii) $\{\psi\} \rightsquigarrow_{\mathcal{T}} \{\psi'\}$ if $\{\psi\} \rightsquigarrow_{\mathcal{T}} \{X\}$ and $\mathcal{T} \models X \sqsubseteq \exists r.\psi'$.

For $\psi \in \mathcal{V}$, we define $\text{ForwardReach}_{\mathcal{T}}(\psi) = \{\psi' \mid \{\psi\} \rightsquigarrow_{\mathcal{T}} \{\psi'\}\}$.

Definition 171 Let \mathcal{T} be a normalised $\mathcal{EL}\mathcal{H}^r$ -terminology \mathcal{T} and Σ a signature. Additionally, let $\varphi \in \mathbb{N}_C \cup \{\text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_R\}$.

If $\varphi = \text{ran}(r)$, we define

$$\text{Reach}_{\mathcal{T}}(\varphi) := \text{ForwardReach}_{\mathcal{T}}(\varphi) \cup \text{ForwardReach}_{\mathcal{T}}(\text{dom}(r));$$

otherwise, let $\text{Reach}_{\mathcal{T}}(\varphi) := \text{ForwardReach}_{\mathcal{T}}(\varphi)$.

The definition of $\text{Reach}_{\mathcal{T}}(X)$ coincides with the definition of $F_{\mathcal{T}}(C)$ in Section 6.3.1

Lemma 172 Let \mathcal{T} a normalised $\mathcal{EL}\mathcal{H}^r$ -terminology \mathcal{T} and Σ a signature. Additionally, let $\mathcal{V} := \mathbb{N}_C \cup \{\text{dom}(r), \text{ran}(r) \mid r \in \Sigma \cup \text{sig}(\mathcal{T})\}$ and $X \in \mathcal{V}$, let E be an \mathcal{EL}^{\square} -concept, and let $r_1, \dots, r_n \in \mathbb{N}_R$ ($n \geq 0$) such that $\mathcal{T} \models X \sqsubseteq \exists r_1 \dots \exists r_n.E$. Then there exists $Y \in \text{ForwardReach}_{\mathcal{T}}(X)$ such that $\mathcal{T} \models Y \sqsubseteq E$.

Proof By induction on n . For $n = 0$, nothing remains to be shown as $X \in \text{ForwardReach}_{\mathcal{T}}(X)$.

Let now $n \geq 1$. It then follows from Lemma 169 that there exists an axiom $Y \boxtimes \exists s.Z \in \mathcal{T}$ ($\boxtimes \in \{\sqsubseteq, \equiv\}$) such that $\mathcal{T} \models X \sqsubseteq Y$, $\mathcal{T} \models s \sqsubseteq r$, and $\mathcal{T} \models Z \sqsubseteq \exists r_2 \dots \exists r_n.E$. Furthermore, we obtain from the induction hypothesis that there exists $Z' \in \text{ForwardReach}_{\mathcal{T}}(Z)$ such that $\mathcal{T} \models Z' \sqsubseteq E$. As $\{X\} \rightsquigarrow_{\mathcal{T}} \{Z\}$, it is then easy to see that $Z' \in \text{ForwardReach}_{\mathcal{T}}(X)$. \square

Lemma 173 Let \mathcal{T} be a normalised $\mathcal{EL}\mathcal{H}^r$ -terminology and Σ a signature. Additionally, let $x_X \in \mathcal{V}$ with $X \in \mathbb{N}_C \cup \{\text{dom}(r) \mid r \in \mathbb{N}_R\} \cup \{\text{ran}(r) \mid r \in \mathbb{N}_R\}$ and let D be an \mathcal{EL}^{\square} -concept. Then $\mathcal{T} \models X \sqsubseteq \exists u.D$ iff there exists $x_Y \in \text{Reach}_{\mathcal{T}}(x_X)$ such that $\mathcal{T} \models Y \sqsubseteq D$.

Proof The direction “ \Leftarrow ” follows immediately from the definition of the set $\text{Reach}_{\mathcal{T}}(x_X)$ and of the relation $\rightsquigarrow_{\mathcal{T}}$.

For the direction “ \Rightarrow ” assume that $\mathcal{T} \models X \sqsubseteq \exists u.D$ holds. We now distinguish between the following cases.

If $X = A$, it follows from Lemma 167 that there exist $n \geq 0$ and $r_1, \dots, r_n \in \mathbb{N}_R$ such that $\mathcal{T} \models A \sqsubseteq \exists r_1 \dots \exists r_n.D$. Thus, by Lemma 172 there exists $x_Y \in \text{ForwardReach}_{\mathcal{T}}(x_A) \subseteq \text{Reach}_{\mathcal{T}}(x_A)$ such that $\mathcal{T} \models Y \sqsubseteq D$.

If $X = \text{ran}(r)$, it follows from Lemma 167 that there exist $n \geq 0$ and $r_1, \dots, r_n \in \mathbb{N}_R$ such that either

- $\mathcal{T} \models \text{ran}(r) \sqsubseteq \exists r_1 \dots \exists r_n . D$, or
- $\mathcal{T} \models \text{dom}(r) \sqsubseteq \exists r_1 \dots \exists r_n . D$.

In the former case we can prove the statement of the lemma analogously to the case for $X = A$. In the latter case we observe that by Lemma 172 there exists $x_Y \in \text{ForwardReach}_{\mathcal{T}}(x_{\text{dom}(r)}) \subseteq \text{Reach}_{\mathcal{T}}(x_X)$ such that $\mathcal{T} \models Y \sqsubseteq D$.

The case for $X = \text{dom}(r)$ can be shown analogously. \square

It is shown in [23] that $\text{lhsWtn}_{\Sigma}^{\text{ran}}(\mathcal{T}_1, \mathcal{T}_2) = \text{lhsWtn}_{\Sigma}^{\text{Ran}}(\mathcal{T}_1, \mathcal{T}_2)$.

Lemma 174 *Let \mathcal{T} be a normalised $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminology, let X be a concept name, let $\exists r.D$ be an $\mathcal{E}\mathcal{L}$ -concept such that $\mathcal{T} \models X \sqsubseteq \exists r.D$. Then there exists an axiom $Y \bowtie \exists s.Z \in \mathcal{T}$ ($\bowtie \in \{\sqsubseteq, \equiv\}$) such that $\mathcal{T} \models X \sqsubseteq Y$, $\mathcal{T} \models s \sqsubseteq r$ and $\mathcal{T} \models Z \sqsubseteq D$.*

Proof Let Δ be the derivation of the inclusion $X \sqsubseteq \exists r.D$ w.r.t. \mathcal{T} . We assume now towards a contradiction that there does not exist an application of the (DEFL), (PDEFL) or (SUB) rule w.r.t. an axiom $Y \bowtie \exists r.Z \in \mathcal{T}$ in Δ . It is then easy to see that no concept of the form $\exists r.E$ occurs on the left-hand side of any inclusion in Δ . We can infer that no concept of the form $\exists r.E$ occurs on the right-hand side of any inclusion in Δ as well, i.e. Δ is not a derivation of the inclusion $A \sqsubseteq \exists r.D$, which contradicts our assumption. \square

Lemma 175 *Let \mathcal{T}_1 and \mathcal{T}_2 be normalised $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies, let Σ be a signature. Moreover, let $X_1, X_2 \in \mathbb{N}_C \cup \text{dom}(r) \cup \text{ran}(r)$, $r \in \mathbb{N}_R$.*

Then: If $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \text{Ran}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$ holds, then $\mathcal{T}_2 \models X_2 \sqsubseteq C$ holds for every $C \in \mathcal{E}\mathcal{L}_{\Sigma}$ such that $\mathcal{T}_1 \models X_1 \sqsubseteq C$.

Proof We prove by induction on the structure of C that for every $X_1, X_2 \in \mathbb{N}_C \cup \text{dom}(r) \cup \text{ran}(r) \cup \{\top\}$ with $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \text{Ran}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$ and $\mathcal{T}_1 \models X_1 \sqsubseteq C$ it holds that $\mathcal{T}_2 \models X_2 \sqsubseteq C$.

Let $C \in \mathcal{E}\mathcal{L}_{\Sigma}$ and let $\varphi_1, \varphi_2 \in \mathbb{N}_C$ such that $(X_1, X_2) \in S$ and $\mathcal{T}_1 \models \varphi \sqsubseteq C$. We now distinguish between the following cases.

- $C = \top$: then we are done, since $\mathcal{T}_2 \models X_2 \sqsubseteq \top$ always holds.
- $C \in \Sigma \cap \mathbb{N}_C$ or $C = \text{dom}(r)$: then we have $\mathcal{T}_2 \models X_2 \sqsubseteq C$ by Definition 154.
- $C = \exists r.E$: then by Lemma 174 there exists an axiom $Y_1 \bowtie_1 \exists s.X_1 \in \mathcal{T}_1$ ($\bowtie_1 \in \{\sqsubseteq, \equiv\}$) such that $\mathcal{T}_1 \models X_1 \sqsubseteq Y_1$, $\mathcal{T}_1 \models \psi_1 \sqsubseteq E$ and $\mathcal{T} \models s \sqsubseteq r$. As S is Σ -subsumer simulation, there exists $Y_2 \bowtie_2 \exists s'.\psi_2 \in \mathcal{T}_2$ with $\mathcal{T}_2 \models X_2 \sqsubseteq Y_2$, $\bowtie_2 \in \{\sqsubseteq, \equiv\}$, $\mathcal{T} \models s' \sqsubseteq r$ and $(X_1, X_2) \in S$. By applying the induction hypothesis we have $\mathcal{T} \models \psi_2 \sqsubseteq E$. Consequently, it holds that $\mathcal{T}_2 \models X_2 \sqsubseteq \exists r.E$.

- $C = C_1 \sqcap C_2$: then we have $\mathcal{T}_1 \models X_1 \sqsubseteq C_i$ for $i \in \{1, 2\}$ and by induction $\mathcal{T}_2 \models X_2 \sqsubseteq C_i$ for $i \in \{1, 2\}$, which finally yields $\mathcal{T}_2 \models X_2 \sqsubseteq C$.

□

Theorem 176 Let $\chi = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{Ran} \rangle$ where \mathcal{T}_1 and \mathcal{T}_2 are two acyclic normalised \mathcal{ELH}^r -terminologies, Σ is a signature and $X_1, X_2 \in \mathbf{N}_C \cup \{ \text{dom}(r), \text{ran}(r) \mid r \in \mathbf{N}_R \}$. Additionally, let

$$\mathbb{M}_{\chi}^{\rightarrow} := \text{COVER}_{\rightarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{Ran}).$$

If $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \text{Ran}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$, then $\mathbb{M}_{\chi}^{\rightarrow}$ is the set of subsumer projection justification under χ .

Proof We first prove that $\mathbb{M}_{\chi}^{\rightarrow}$ is a cover of the set such that $\mathcal{M} \in \mathbb{M}_{\chi}^{\rightarrow}$ is a subsumer projection module under χ . By Theorem 156, we have that proving $\mathbb{M}_{\chi}^{\rightarrow}$ is the set of \mathcal{M} such that subsumer projection module under χ is equal to prove $\mathbb{M}_{\chi}^{\rightarrow} := \{ \mathcal{M} \subseteq \mathcal{T}_2 \mid \langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \text{Ran}}^{\rightarrow} \langle \mathcal{M}, X_2 \rangle \}$ for Ran-subsumption query. .

(i) We show that every $\mathcal{M} \in \mathbb{M}_{\chi}^{\rightarrow}$ is an subsumer projection module under χ .

First, we define the relation $\triangleright_{\mathcal{T}} \subseteq \mathbf{N}_C \times \mathbf{N}_C$ as follows: let \mathcal{T} be an \mathcal{ELH}^r terminology. For $X, Y \in \mathbf{N}_C$ we set $X \triangleright_{\mathcal{T}} Y$ iff there exists $X' \bowtie \exists r.Y \in \mathcal{T}$ with $\mathcal{T} \models X \sqsubseteq X'$ and $\bowtie \in \{ \sqsubseteq, \equiv \}$. Note that $\triangleright_{\mathcal{T}}$ is well-founded as \mathcal{T} is acyclic. The proof now proceeds by induction on $\triangleright_{\mathcal{T}_1}$.

Let $X_1, X_2 \in \mathbf{N}_C$ and let $\mathcal{M} \in \mathbb{M}_{\chi}^{\rightarrow}$. We now distinguish between the following cases.

- If there does not exist $Y \in \mathbf{N}_C$ with $X_1 \triangleright_{\mathcal{T}_1} Y$, let $D \in \mathcal{EL}_{\Sigma}$ such that $\mathcal{T}_1 \models X_1 \sqsubseteq D$. The proof now continues by induction on the structure of D . For $D = \top$ or $D = X$, it immediately follows that $\mathcal{M} \models X_2 \sqsubseteq D$. For $D = B \in \mathbf{N}_C$ and $B \neq X$, there exists $\mathcal{J} \in \text{Just}_{\mathcal{T}_2}(X_2 \sqsubseteq B)$ with $\mathcal{J} \subseteq \mathcal{M}$ (Line 4 in Algorithm 19), i.e. $\mathcal{M} \models X \sqsubseteq D$ holds. Similar with the case when $D = \text{dom}(r)$ and $r \in \mathbf{N}_R$. For $D = D_1 \sqcap D_2$ we obtain $\mathcal{M} \models X_2 \sqsubseteq D_1$ and $\mathcal{M} \models X_2 \sqsubseteq D_2$ by applying the induction hypothesis on D_1 and D_2 . Consequently, it holds that $\mathcal{M} \models X_2 \sqsubseteq D$. Note that D cannot be of the form $\exists r.E$ by Lemma 174.
- Otherwise, let $D \in \mathcal{EL}_{\Sigma}$ such that $\mathcal{T}_1 \models X_1 \sqsubseteq D$. The proof now continues by induction on the structure of D . The cases of $D = \top$, $D = A \in \mathbf{N}_C$, $D = \text{dom}(s), s \in \mathbf{N}_R$ and $D = D_1 \sqcap D_2$ can be proved as above. For D is of the form $\exists s.D'$. there exists $Y \bowtie \exists r.Z \in \mathcal{T}_1$ such that $\mathcal{T}_1 \models X_1 \sqsubseteq Y$, $\mathcal{T}_1 \models r \sqsubseteq s$ and $\mathcal{T}_1 \models Z \sqsubseteq D'$ by Lemma 174. As $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \text{Ran}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$, we have $\mathcal{T}_2 \models X_2 \sqsubseteq \exists s.D'$. Consequently, we have that there exists $\alpha = Y' \bowtie \exists r'.Z' \in \mathcal{T}_2$ such that $\mathcal{T}_2 \models X_2 \sqsubseteq Y'$ and $\mathcal{T}_2 \models r' \sqsubseteq s$ by Lemma 174. and $\alpha \in \mathcal{M}$. Furthermore, there exists $\mathcal{M}_{Z'} \in \text{COVER}_{\rightarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{Ran})$ such that $\mathcal{M}_{Z'} \subseteq \mathcal{M}$. As $X \triangleright_{\mathcal{T}_1} Z'$, it follows from the induction hypothesis $\mathcal{M}_{Z'}$ is a subsumer projection module under $\chi' = \{ \mathcal{T}_1, Z, \Sigma, \mathcal{T}_2, Z', \text{Ran} \}$, i.e. for every $D'' \in \mathcal{EL}_{\Sigma}$, $\mathcal{T}_1 \models Z \sqsubseteq D''$ implies $\mathcal{M}_{Z'} \models Z' \sqsubseteq D''$. Hence, we can infer that $\mathcal{M}_{\exists s.Z} \models X_2 \sqsubseteq \exists r.D'$ (Line 7 in Algorithm 20).

(ii) We show that for every subsumer projection module under χ , there exists $\mathcal{M}' \in \mathbb{M}_{\vec{X}}$ such that $\mathcal{M}' \subseteq \mathcal{M}$.

The proof now proceeds by induction on \triangleright . Let $X_1, X_2 \in \mathbb{N}_{\mathbb{C}}$ and let \mathcal{M} be a subsumer projection module under $\chi = \{\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{Ran}\}$. We now distinguish between the following cases.

- If there does not exist $Y \in \mathbb{N}_{\mathbb{C}}$ with $X_1 \succ_{\mathcal{T}_1} Y$, let $B \in \Sigma$. As \mathcal{M} is a $\langle \mathcal{T}_1, X_1, \Sigma \rangle$ subsumer projection module of \mathcal{T}_2 , there exists $S_B \in \text{Just}_{\mathcal{T}_2}(X_2 \sqsubseteq B)$ with $S_B \subseteq \mathcal{M}$. Consequently, it is easy to see that there exists $\mathcal{M}' \in \mathbb{M}_{\vec{X}}$ with $\mathcal{M}' = \bigcup_{B \in \Sigma} S_B$ as there does not exist $X' \boxtimes \exists s.Y \in \mathcal{T}_2$ with $\mathcal{T}_2 \models X \sqsubseteq X'$ and $\mathcal{T}_2 \models s \sqsubseteq r$.
- Otherwise, $\langle \mathcal{T}_1, X_1 \rangle \sim_{\vec{\Sigma}, \text{Ran}} \langle \mathcal{T}_2, X_2 \rangle$. Similarly to the previous case, for every $B \in \Sigma$ there exists $S_B \in \text{Just}_{\mathcal{T}_2}(X \sqsubseteq B)$ with $S_B \subseteq \mathcal{M}$.

Now let $\{Y \boxtimes \exists r.Z \in \mathcal{T}_1 \mid r \in \Sigma, \mathcal{T}_1 \models X_1 \sqsubseteq Y \text{ and } \mathcal{T}_1 \models s \sqsubseteq r\} = \{Y_1 \boxtimes \exists r_1.Z_1, \dots, Y_n \boxtimes \exists r_n.Z_n\}$ and let $1 \leq i \leq n$. Thus, as $\langle \mathcal{T}_1, X_1 \rangle \sim_{\vec{\Sigma}, \text{Ran}} \langle \mathcal{T}_2, X_2 \rangle$, by the definition of subsumer simulation (Definition 154), there exists $Y'_i \boxtimes \exists s_i.Z'_i \in \mathcal{M}$ such that $\mathcal{M} \models X \sqsubseteq Y'_i$ and $\mathcal{M} \models s_i \sqsubseteq r_i$ with $\langle \mathcal{T}_1, Z \rangle \sim_{\vec{\Sigma}, \text{Ran}} \langle \mathcal{T}_2, Z'_i \rangle$ holds. Consequently, there exists $S_{Y'_i} \in \text{Just}_{\mathcal{T}_2}(X \sqsubseteq Y'_i)$ with $S_{Y'_i} \subseteq \mathcal{M}$ and $S_{r'_i} \in \text{Just}_{\mathcal{T}_2}(s_i \sqsubseteq r_i)$ with $S_{r'_i} \subseteq \mathcal{M}$. Moreover, as $(Z, Z'_i) \in S$, we have that \mathcal{M} is a $\langle Z'_i, \Sigma \rangle$ -subsumer module of \mathcal{T}_2 . It follows from the induction hypothesis that there exists $\mathcal{M}'_{Z'_i} \in \mathbb{M}_{\vec{Z}'_i} := \text{COVER}_{\rightarrow}(\mathcal{T}_1, Z_i, \Sigma, \mathcal{T}_2, Z'_i)$ with $\mathcal{M}'_{Z'_i} \subseteq \mathcal{M}$. Let

$$\mathcal{M}' := \bigcup_{B \in \Sigma} S_B \cup \bigcup_{i=1}^n (S_{Y'_i} \cup S_{r'_i} \cup \{Y'_i \boxtimes \exists s_i.Z'_i\} \cup \mathcal{M}'_{Z'_i}).$$

It is easy to see that $\mathcal{M}' \in \mathbb{M}_{\vec{X}}$ and $\mathcal{M}' \subseteq \mathcal{M}$.

Then we prove that for any $\mathcal{M} \in \mathbb{M}_{\vec{X}}$, \mathcal{M} is a subsumer projection justification under χ . We assume that \mathcal{M} is a subsumer projection justification under χ and there exist a redundant axiom $\alpha \in \mathcal{M}$, such that $\mathcal{M} \setminus \{\alpha\}$ is still a subsumer module under χ . Once we go through all possible case when \mathbb{S} is \otimes to $\mathbb{M}_{\vec{X}}$ such that $\alpha \in S$ and $S \in \mathbb{S}$, we have that if α is removed from S , the subsumer simulation does not hold anymore. By Theorem 156, we have that there exists $D \in \mathcal{E}\mathcal{L}_{\Sigma}^{\text{ran}}$, such that $\mathcal{T}_1 \models X_1 \sqsubseteq D$ does not imply $\mathcal{T}_2 \models X_2 \sqsubseteq D$. So, \mathcal{M} is not a subsumer projection justification under χ , which contradict our assumption. \square

Lemma 177 *Let \mathcal{T}_1 and \mathcal{T}_2 be two acyclic normalised $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies, Σ is a signature and $X, X' \in \mathbb{N}_{\mathbb{C}} \cup \{\text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_{\mathbb{R}}\}$. If $\langle \mathcal{T}_1, X \rangle \sim_{\vec{\Sigma}, \text{Ran}^{\square, u}} \langle \mathcal{T}_2, X' \rangle$, then $\mathcal{T}_2 \models X' \sqsubseteq C$ for every $C \in \mathcal{E}\mathcal{L}_{\Sigma}^{\square, u}$ such $\mathcal{T}_1 \models X \sqsubseteq C$.*

Proof Let $\langle \mathcal{T}_1, X \rangle \sim_{\vec{\Sigma}, \text{Ran}^{\square, u}} \langle \mathcal{T}_2, X' \rangle$ and let C be an $\mathcal{E}\mathcal{L}_{\Sigma}^{\square, u}$ -concept C with $\mathcal{T}_1 \models X \sqsubseteq C$. The proof now proceeds by induction on the structure of C .

- $C = \top$: then we are done, since $\mathcal{T}_2 \models A \sqsubseteq \top$ always holds.
- $C \in \mathbb{N}_C$ or $C = \text{dom}(r)$: then we have $\mathcal{T}_1 \models X \sqsubseteq C$ implies $\mathcal{T}_2 \models X' \sqsubseteq C$ by Definition 154.
- $C = C_1 \sqcap \dots \sqcap C_n$: then we have $\mathcal{T}_1 \models X \sqsubseteq C_i$ for every $1 \leq i \leq n$ and by induction $\mathcal{T}_2 \models X' \sqsubseteq C_i$, which finally yields $\mathcal{T}_2 \models X' \sqsubseteq C$.
- $C = \exists R.D$ with $R = r_1 \sqcap \dots \sqcap r_n$ ($n \geq 1$) and D an $\mathcal{EL}_{\Sigma}^{\sqcap}$ -concept: let $r \in \mathbb{N}_R$ such that $\mathcal{T}_1 \models r \sqsubseteq r_i$ for every $1 \leq i \leq n$ and $\mathcal{T}_1 \models X \sqsubseteq \exists r.D$ (cf. Lemma 168). Then by Lemmas 169 there exists a $Z \bowtie \exists s.Y \in \mathcal{T}_1$ ($\bowtie \in \{\sqsubseteq, \equiv\}$) such that $\mathcal{T}_1 \models X \sqsubseteq Z$, $\mathcal{T}_1 \models s \sqsubseteq r$ and $\mathcal{T}_1 \models Y \sqsubseteq D$. By definition 154, there exists a $Z' \bowtie \exists s'.Y' \in \mathcal{T}_2$ ($\bowtie \in \{\sqsubseteq, \equiv\}$) such that $\mathcal{T}_2 \models X' \sqsubseteq Z'$, $\mathcal{T}_2 \models s' \sqsubseteq r_i$ for every $i \in [1, n]$ and $\mathcal{T}_2 \models Y' \sqsubseteq D$. Thus we get $\mathcal{T}_2 \models Y' \sqsubseteq D$ by applying the induction hypothesis. Hence we have $\mathcal{T}_2 \models X' \sqsubseteq \exists r.D$.

Let now $X \in (\mathbb{N}_C \cap \Sigma) \cup \{\text{dom}(r) \mid r \in \Sigma\} \cup \{\text{ran}(r) \mid r \in \Sigma\}$, and let D be an $\mathcal{EL}_{\Sigma}^{\sqcap, u}$ -concept such that $\mathcal{T}_1 \models X \sqsubseteq D$. Now, let D' be the $\mathcal{EL}_{\Sigma}^{\sqcap, u}$ -concept of the form $D'_0 \sqcap \prod_{i=1}^n \exists u.D_i^u$, where D'_0 and D_i^u ($1 \leq i \leq n$) are $\mathcal{EL}_{\Sigma}^{\sqcap}$ -concepts, such that $\models D \equiv D'$. It then follows from the previous parts of the proof that $\mathcal{T}_2 \models X \sqsubseteq D'_0$ holds. Let now $1 \leq i \leq n$. It then follows from Lemma 173 that there exists $x_Y \in \text{Reach}_{\mathcal{T}_1}(x_X)$ such that $\mathcal{T}_1 \models Y \sqsubseteq D_i^u$. By definition of $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{L}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$ (Definition 154), there is $x_{Y'} \in \text{Reach}_{\mathcal{T}_2}(x_{X'})$ such that $x_Y \xrightarrow{f}_{\Sigma} x_{Y'}$. We thus obtain $\mathcal{T}_2 \models Y' \sqsubseteq D_i^u$ from the previous parts of the proof, and it follows that $\mathcal{T}_2 \models X \sqsubseteq \exists u.D_i^u$ holds by Lemma 173. Overall, we can conclude that $\mathcal{T}_2 \models X \sqsubseteq D'$, i.e. $\mathcal{T}_2 \models X \sqsubseteq D$. \square

Lemma 178 *If $\mathcal{T}_1 \sim_{\Sigma, \text{Ran}^{\sqcap, u}}^{\rightarrow} \mathcal{T}_2$, then $\text{lhsWtn}_{\Sigma}^{\text{Ran}^{\sqcap, u}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$.*

Proof Let D be an $\mathcal{EL}_{\Sigma}^{\sqcap, u}$ -concept such that $\mathcal{T}_1 \models X_1 \sqsubseteq D$. Now, let D' be the $\mathcal{EL}_{\Sigma}^{\sqcap, u}$ -concept of the form $D'_0 \sqcap \prod_{i=1}^n \exists u.D_i^u$, where D'_0 and D_i^u ($1 \leq i \leq n$) are $\mathcal{EL}_{\Sigma}^{\sqcap}$ -concepts, such that $\models D \equiv D'$. It then follows from Lemma 177 that $\mathcal{T}_2 \models X \sqsubseteq D'_0$ holds. Let now $1 \leq i \leq n$. It then follows from Lemma 173 that there exists $Y \in \text{Reach}_{\mathcal{T}_1}(x_X)$ such that $\mathcal{T}_1 \models Y \sqsubseteq D_i^u$. By definition of $\mathcal{T}_1 \sim_{\Sigma, \text{Ran}}^{\rightarrow} \mathcal{T}_2$, there is $Y' \in \text{Reach}_{\mathcal{T}_2}(X')$ such that $Y \xrightarrow{f}_{\Sigma} Y'$. We thus obtain $\mathcal{T}_2 \models Y' \sqsubseteq D_i^u$ from the previous parts of the proof, and it follows that $\mathcal{T}_2 \models X \sqsubseteq \exists u.D_i^u$ holds by Lemma 173. Overall, we can conclude that $\mathcal{T}_2 \models X \sqsubseteq D'$, i.e. $\mathcal{T}_2 \models X \sqsubseteq D$. \square

Lemma 179 *Let $\mathcal{T}_1, \mathcal{T}_2$ be two normalised \mathcal{ELH}^r -terminology and Σ a signature. Additionally, let $X \in \mathbb{N}_R \cap \text{sig}(\mathcal{T}_1) \cap \{\text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_R\}$ and $X \in \mathbb{N}_R \cap \text{sig}(\mathcal{T}_1)$ and $X' \in \mathbb{N}_R \cap \text{sig}(\mathcal{T}_1) \cap \{\text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_R\}$ and $X \in \mathbb{N}_R \cap \text{sig}(\mathcal{T}_2)$ such that $\mathcal{T}_1 \models X \sqsubseteq C$ entails $\mathcal{T}_2 \models X' \sqsubseteq C$ for every $\mathcal{EL}_{\Sigma}^{\sqcap}$ -concept C . Moreover, let $A \bowtie \exists r.Y \in \mathcal{T}_1$ ($\bowtie \in \{\equiv, \sqsubseteq\}$) such that there exists $s \in \Sigma$ with $\mathcal{T}_1 \models r \sqsubseteq s$ and $\mathcal{T}_1 \models X \sqsubseteq A$.*

Then there exists $A' \bowtie \exists r'.Y' \in \mathcal{T}_2$ such that the following holds:

- $\mathcal{T}_2 \models X' \sqsubseteq A'$;

- $\mathcal{T}_2 \models r' \sqsubseteq s$ for every $s \in \Sigma$ with $\mathcal{T}_2 \models s \sqsubseteq r$; and
- if $\mathcal{T}_1 \models Y \sqsubseteq C$, then $\mathcal{T}_2 \models Y' \sqsubseteq C$ for every $\mathcal{E}\mathcal{L}_\Sigma^\square$ -concept C .

Proof Let $S = \{s \in \Sigma \mid \mathcal{T}_1 \models s \sqsubseteq r\}$ and let

$$\bigcup_{i=1}^n \{A'_i \sqsubseteq \exists r'_i.Y'_i\} = \{A' \sqsubseteq \exists r'.X' \mid \mathcal{T}_2 \models X' \sqsubseteq A'\},$$

$$A' \bowtie \exists r'.Y' \in \mathcal{T}_2, \bowtie \in \{\sqsubseteq, \equiv\}, \forall s \in S: \mathcal{T}_2 \models r' \sqsubseteq s\}.$$

Let $R = \prod_{s \in S} s$. As $\mathcal{T}_1 \models A \sqsubseteq \exists R.\top$, we have $\mathcal{T}_2 \models A \sqsubseteq \exists R.\top$, and therefore $n \geq 1$ by Lemmas 168 and 169. If we assume for every $1 \leq i \leq n$ that there exists a $\mathcal{E}\mathcal{L}_\Sigma^\square$ -concept C_i with $\mathcal{T}_1 \models \text{ran}(r_i) \sqcap Y \sqsubseteq C_i$ and $\mathcal{T}_2 \not\models \text{ran}(r'_i) \sqcap X'_i \sqsubseteq C_i$, it would follow that $\mathcal{T}_1 \models X \sqsubseteq \exists R.(C_1 \sqcap \dots \sqcap C_n)$ and hence, $\mathcal{T}_2 \models X \sqsubseteq \exists R.(C_1 \sqcap \dots \sqcap C_n)$. By Lemmas 168 and 169 we could infer that there exists $1 \leq j \leq n$ such that $\mathcal{T}_2 \models A'_j \sqsubseteq \exists r'_j.X'_j$ and $\mathcal{T}_2 \models \text{ran}(r'_j) \sqcap X'_j \sqsubseteq C_j$ for every $1 \leq i \leq n$, which contradicts our assumption. \square

Lemma 180 *If $\text{lhsWtn}_{\Sigma}^{\text{Ran}^{\square,u}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$, then $\mathcal{T}_1 \sim_{\Sigma, \text{Ran}^{\square,u}}^{\rightarrow} \mathcal{T}_2$.*

Proof Let

$$S = \{(X, X') \mid \forall C \in \mathcal{E}\mathcal{L}_\Sigma^\square: \mathcal{T}_1 \models X \sqsubseteq C \implies \mathcal{T}_2 \models X' \sqsubseteq C\}.$$

Clearly, $(X, X) \in S$ for every $X \in (\mathbf{N}_C \cap \Sigma) \cup \{\text{dom}(r), \text{ran}(r) \mid r \in \mathbf{N}_R\}$. We now show that S is a $\langle \Sigma, \text{Ran}^{\square,u} \rangle$ -simulation:

- Let $(X, X') \in S$ with $\mathcal{T}_1 \models X \sqsubseteq Z$, and thus by definition of S also $\mathcal{T}_2 \models X' \sqsubseteq Z$, which satisfying Condition $(S_{\mathbf{N}_C}^{\rightarrow})$ of Definition 154.
- Let $(X, X') \in S$ with $X \bowtie_1 \exists r.Y \in \mathcal{T}_1 \bowtie_1 \in \{\sqsubseteq, \equiv\}$ such that $\mathcal{T}_1 \models r \sqsubseteq s$ for some $s \in \Sigma$. Then by 179, there exists $X'_2 \bowtie_2 \exists r'.Y' \in \mathcal{T}_2$ with $\bowtie_2 \in \{\sqsubseteq, \equiv\}$ such that $\mathcal{T}_2 \models X_2 \sqsubseteq X'_2$ and for every $s \in \Sigma$ with $\mathcal{T}_1 \models r \sqsubseteq s$, and $\mathcal{T}_1 \models Y \sqsubseteq C$ implies $\mathcal{T}_2 \models Y' \sqsubseteq C$ for all $\mathcal{E}\mathcal{L}_\Sigma^\square$ -concepts C . But then $(Y, Y') \in S$ by definition, satisfying Condition $(S_{\exists, \text{Ran}^{\square,u}}^{\rightarrow})$ of Definition 154 overall.

Let now $X \in (\mathbf{N}_C \cap \Sigma) \cup \{\text{dom}(r) \mid r \in \Sigma\} \cup \{\text{ran}(r) \mid r \in \Sigma\}$. We assume towards a contradiction that there exists $x_Y \in \text{Reach}_{\mathcal{T}_1}(x_X)$ such that $(x_Y, x_{Y'}) \notin S$ for every $x_{Y'} \in \text{Reach}_{\mathcal{T}_2}(x_X)$. Hence, for every $x_{Y'} \in \text{Reach}_{\mathcal{T}_2}(x_X)$ there exists an $\mathcal{E}\mathcal{L}_\Sigma^\square$ -concept $C_{Y'}$ such that $\mathcal{T}_1 \models Y \sqsubseteq C_{Y'}$ and $\mathcal{T}_2 \not\models Y' \sqsubseteq C_{Y'}$. Let now $\mathbf{D} = \prod_{x_{Y'} \in \text{Reach}_{\mathcal{T}_2}(x_X)} C_{Y'}$. We can then infer that $\mathcal{T}_1 \models Y \sqsubseteq \mathbf{D}$ and $\mathcal{T}_2 \not\models Y' \sqsubseteq \mathbf{D}$ for every $x_{Y'} \in \text{Reach}_{\mathcal{T}_2}(x_X)$. By Lemma 173 it holds that $\mathcal{T}_1 \models X \sqsubseteq \exists u.\mathbf{D}$ but $\mathcal{T}_2 \not\models X \sqsubseteq \exists u.\mathbf{D}$, which contradicts with $\text{lhsWtn}_{\Sigma}^{\text{Ran}^{\square,u}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$. \square

Theorem 181 Let $\chi = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{Ran}^{\square, u} \rangle$, \mathcal{T}_1 and \mathcal{T}_2 be two acyclic normalised $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies, and let Σ be a signature. Additionally, let $X \in (\mathbb{N}_C \cap \Sigma) \cup \{ \text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_R \}$, and let $\mathbb{M}_X^{\rightarrow} := \text{COVER}_{\rightarrow} \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{Ran}^{\square, u}$.

Then: If $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \text{Ran}^{\square, u}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$ holds, then $\mathbb{M}_X^{\rightarrow}$ is the set of all subsumer projection justifications under χ .

Proof The proof is similar to the proof of Lemma 176 (by using Lemma 179). □

Theorem 156 If $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \mathcal{L}}^{\rightarrow} \langle \mathcal{T}_2, X_2 \rangle$, then $\mathcal{T}_2 \models X_2 \sqsubseteq D$ for every $D \in \mathcal{L}_{\Sigma}$ such that $\mathcal{T}_1 \models X_1 \sqsubseteq D$.

Proof It follows from Lemma 175 (Case Ran) and Lemma 177 (Case $\text{Ran}^{\square, u}$). □

Lemma 182 Let \mathcal{T}_1 and \mathcal{T}_2 be two normalised $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies and Σ be a signature. Additionally, let $X \in \mathbb{N}_R \cap \text{sig}(\mathcal{T}_1) \cap \{ \text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_R \}$ and $X' \in \mathbb{N}_R \cap \text{sig}(\mathcal{T}_2) \cap \{ \text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_R \}$ and $X \in \mathbb{N}_R \cap \text{sig}(\mathcal{T}_1)$ be such that $\mathcal{T}_1 \models X \sqsubseteq C$ entails $\mathcal{T}_2 \models X' \sqsubseteq C$ for every $\mathcal{E}\mathcal{L}_{\Sigma}^{\text{ran}}$ -concept C . Moreover, let $A \bowtie \exists r.Y \in \mathcal{T}_1$ ($\bowtie \in \{ \equiv, \sqsubseteq \}$) such that there exists $s \in \Sigma$ with $\mathcal{T}_1 \models r \sqsubseteq s$ and $\mathcal{T}_1 \models X \sqsubseteq A$.

Then there exists $A' \bowtie \exists r'.Y' \in \mathcal{T}_2$ such that the following holds

- $\mathcal{T}_2 \models X' \sqsubseteq A'$;
- $\mathcal{T}_2 \models r' \sqsubseteq s$; and
- If $\mathcal{T}_1 \models Y \sqsubseteq C$, then $\mathcal{T}_2 \models Y' \sqsubseteq C$ for every $\mathcal{E}\mathcal{L}_{\Sigma}^{\text{ran}}$ -concept C .

Proof The lemma can be proved analogously to Lemma 179.

Let $S = \{ s \in \Sigma \mid \mathcal{T}_1 \models s \sqsubseteq r \}$ and let

$$\bigcup_{i=1}^n \{ A'_i \sqsubseteq \exists r'_i.Y'_i \} = \{ A' \sqsubseteq \exists r'.X' \mid \mathcal{T}_2 \models X' \sqsubseteq A' \},$$

$$A' \bowtie \exists r'.Y' \in \mathcal{T}_2, \bowtie \in \{ \sqsubseteq, \equiv \}, \forall s \in S: \mathcal{T}_2 \models r' \sqsubseteq s \}.$$

Let $R = \prod_{s \in S} s$. As $\mathcal{T}_1 \models A \sqsubseteq \exists R.T$, we have $\mathcal{T}_2 \models A \sqsubseteq \exists R.T$, and therefore $n \geq 1$ by Lemmas 168 and 169. If we assume for every $1 \leq i \leq n$ that there exists a $\mathcal{E}\mathcal{L}_{\Sigma}^{\square}$ -concept C_i with $\mathcal{T}_1 \models \text{ran}(r_i) \sqcap Y \sqsubseteq C_i$ and $\mathcal{T}_2 \not\models \text{ran}(r'_i) \sqcap X'_i \sqsubseteq C_i$, it would follow that $\mathcal{T}_1 \models X \sqsubseteq \exists R.(C_1 \sqcap \dots \sqcap C_n)$ and hence, $\mathcal{T}_2 \models X \sqsubseteq \exists R.(C_1 \sqcap \dots \sqcap C_n)$. By Lemmas 168 and 169 we could infer that there exists $1 \leq j \leq n$ such that $\mathcal{T}_2 \models A'_j \sqsubseteq \exists r'_j.X'_j$ and $\mathcal{T}_2 \models \text{ran}(r'_j) \sqcap X'_j \sqsubseteq C_j$ for every $1 \leq i \leq n$, which contradicts our assumption. □

Theorem 155 We have the following conclusions:

- $\mathcal{T}_1 \sim_{\Sigma, \text{ran}}^{\rightarrow} \mathcal{T}_2$ iff $\text{lhsWtn}_{\Sigma}^{\text{ran}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$;
- $\mathcal{T}_1 \sim_{\Sigma, \text{Ran}}^{\rightarrow} \mathcal{T}_2$ iff $\text{lhsWtn}_{\Sigma}^{\text{Ran}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$;
- $\mathcal{T}_1 \sim_{\Sigma, \text{Ran}^{\square, u}}^{\rightarrow} \mathcal{T}_2$ iff $\text{lhsWtn}_{\Sigma}^{\text{Ran}^{\square, u}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$.

Proof We first prove $\mathcal{T}_1 \sim_{\Sigma, \text{Ran}^{\square, u}}^{\rightarrow} \mathcal{T}_2$ iff $\text{lhsWtn}_{\Sigma}^{\text{Ran}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$, which follows from Lemma 178 and Lemma 180. Then $\mathcal{T}_1 \sim_{\Sigma, \text{Ran}}^{\rightarrow} \mathcal{T}_2$ iff $\text{lhsWtn}_{\Sigma}^{\text{Ran}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$ can be proved analogously as above using Lemma 182. As $\text{lhsWtn}_{\Sigma}^{\text{ran}}(\mathcal{T}_1, \mathcal{T}_2) = \text{lhsWtn}_{\Sigma}^{\text{Ran}}(\mathcal{T}_1, \mathcal{T}_2)$, we have that $\mathcal{T}_1 \sim_{\Sigma, \text{ran}}^{\rightarrow} \mathcal{T}_2$ iff $\text{lhsWtn}_{\Sigma}^{\text{ran}}(\mathcal{T}_1, \mathcal{T}_2) = \emptyset$. \square

Theorem 157 Let \mathcal{T}_1 and \mathcal{T}_2 be acyclic, normalised $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies, and let Σ be a signature. Additionally, let $\chi = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L} \rangle$ and $\mathbb{M} := \text{COVER}_{\rightarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L})$ computed by Algorithm 22. If $\mathcal{T}_1 \sim_{\Sigma, \mathcal{L}}^{\rightarrow} \mathcal{T}_2$, then \mathbb{M} is the set of all subsumer projection justifications under χ .

Proof It follows from Theorem 176 (Case Ran) and Theorem 181 (Case $\text{Ran}^{\square, u}$). \square

6.4.4 Proof Regarding Subsumee Projection Justification

Lemma 183 Let \mathcal{T} be a normalised and acyclic $\mathcal{E}\mathcal{L}\mathcal{H}$ -terminology, let Σ be a signature, and let $X \in \mathbb{N}_{\mathbb{C}}$ be pseudo-primitive in \mathcal{T} . Then X is not complex Σ -entailed w.r.t. \mathcal{T} .

Proof As X is pseudo-primitive in \mathcal{T} , we have $\text{non-conj}_{\mathcal{T}}(X) = \{X\}$. We can infer that there cannot exist $B \in \Sigma$ with $\mathcal{T} \models B \sqsubseteq Y$ and $\mathcal{T} \not\models B \sqsubseteq X$. Additionally, we observe that no axiom of the form $X \equiv \exists r.Y$ is contained in \mathcal{T} , which implies that X is not complex Σ -entailed w.r.t. \mathcal{T} . \square

Lemma 184 Let \mathcal{T}_1 and \mathcal{T}_2 be normalised and acyclic $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies, and let Σ be a signature. Then: if $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2, X_2 \rangle$ then $\mathcal{T}_2 \models \text{ran}(\zeta) \sqcap C \sqsubseteq X_2$ holds for every $C \in \mathcal{E}\mathcal{L}_{\Sigma}$, $\mathcal{T}_1 \models \text{ran}(\zeta) \sqcap C \sqsubseteq X_1$.

Proof Let C be an $\mathcal{E}\mathcal{L}_{\Sigma}$ -concept and let $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \text{ran}}^{\leftarrow} \langle \mathcal{T}_2, X_2 \rangle \zeta$ with $X_1, X_2 \in \mathbb{N}_{\mathbb{C}}$ such that $\mathcal{T}_1 \models C \sqsubseteq X_1$ if $\zeta = \varepsilon$, or $\mathcal{T}_1 \models \text{ran}(r) \sqcap C \sqsubseteq X_1$ if $\zeta = r$. We distinguish these three cases, which are mutually exclusive as \mathcal{T}_1 is a terminology and prove by introduction:

- X is pseudo-primitive in \mathcal{T}_1 : it follows from Lemma 34 that there exists $Z \in \Sigma^{(\text{ran}, \zeta)}$ such that $\mathcal{T}_1 \models Z \sqsubseteq X_1$. It follows from Item ($S_{\mathbb{N}_{\mathbb{C}}}^{\leftarrow, \mathcal{L}}$) of Definition 160 that $\mathcal{T}_2 \models Z \sqsubseteq Y_2$ for every $Y_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$ with $\mathcal{T}_2 \not\models \text{ran}(\zeta) \sqsubseteq Y_2$ if $\zeta \neq \varepsilon$. Hence, $\mathcal{T}_2 \models \text{ran}(\zeta) \sqcap D \sqsubseteq X_2$ by Lemma 34.

- there exists an axiom of the form $X_1 \equiv \exists s.B_1$ in \mathcal{T}_1 : By Lemma 34 we have that either a1) there exists $Z \in \Sigma^{(\text{ran}, \zeta)}$ such that $\mathcal{T}_1 \models Z \sqsubseteq X_1$, or a2) there exists a conjunct $\exists t.C'$ of C such that $\mathcal{T}_1 \models t \sqsubseteq s$ and $\mathcal{T}_1 \models \text{ran}(t) \sqcap C' \sqsubseteq B_1$. For a1), we proceed as above and obtain $\mathcal{T}_2 \models \text{ran}(\zeta) \sqcap C \sqsubseteq X_2$. Consider a2). As $\mathcal{T}_1 \models \text{ran}(t) \sqcap C' \sqsubseteq B_1$ and C' is an \mathcal{EL}_Σ -concept, it holds that B is (Σ, ran, t) -entailed by Definition 158. Let $Y_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$. As $t \in \mathbb{N}_R \cap \Sigma$ with $\mathcal{T}_1 \models t \sqsubseteq s$, we obtain by Item $(S_{\exists}^{\leftarrow, \mathcal{L}})$ of Definition 160 that either i) $\mathcal{T}_2 \models \text{dom}(t) \sqsubseteq Y_2$, ii) $\mathcal{T}_2 \models \text{ran}(r) \sqsubseteq Y_2$, for $\zeta = r$, or iii) there exists $Y_2 \equiv \exists r'.B_2 \in \mathcal{T}_2$, such that $\mathcal{T}_2 \models t \sqsubseteq r'$ and $(B_1, B_2, t) \in S$. For i), it implies that $\mathcal{T}_2 \models \exists t.D' \sqsubseteq Y'$. For ii), we obtain $\mathcal{T}_2 \models \text{ran}(r) \sqsubseteq Y_2$. For iii), since $\mathcal{T}_1 \models \text{ran}(t) \sqcap D' \sqsubseteq B$, and D' is an \mathcal{EL}_Σ -concept, the induction hypothesis yields that $\mathcal{T}_2 \models \text{ran}(t) \sqcap D' \sqsubseteq B_2$. Then $\mathcal{T}_2 \models \exists t.(\text{ran}(t) \sqcap D') \sqsubseteq \exists t.B_2$ and, thus, $\mathcal{T}_2 \models \exists t.D' \sqsubseteq Y_2$. For $\zeta = \varepsilon$, we have that $\mathcal{T}_2 \models \exists t.D' \sqsubseteq Y_2$ for every $Y_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$. Hence, $\mathcal{T}_2 \models \exists t.D' \sqsubseteq X_2$. For $\zeta = r$, we obtain that $\mathcal{T}_2 \models \exists t.D' \sqsubseteq Y_2$ or $\mathcal{T}_2 \models \text{ran}(r) \sqsubseteq Y_2$ for every $Y_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$. Hence, $\mathcal{T}_2 \models \text{ran}(r) \sqcap \exists t.D' \sqsubseteq X_2$.
- there exists an axiom of the form $X \equiv X_1 \sqcap \dots \sqcap X_n$ in \mathcal{T}_1 : By Item $(S_{\sqcap}^{\leftarrow, \mathcal{L}})$ of Definition 160 we have that b1) for $\zeta = r$: $\mathcal{T}_2 \models \text{ran}(r) \sqsubseteq Y_2$ or there exists $i \in \{1, \dots, n\}$ such that $\mathcal{T}_1 \not\models \text{ran}(r) \sqsubseteq X_i$ and $(x_{X_i}, x_{Y'}, \varepsilon) \in \leftarrow^{\text{ran}}$, or b2) for $\zeta = \varepsilon$: $(x_{X_i}, x_{Y'}, \varepsilon) \in \leftarrow^{\text{ran}}$ for some $i \in \{1, \dots, n\}$. For b1), we have that $\mathcal{T}_2 \models \text{ran}(r) \sqsubseteq Y_2$, or, since $\mathcal{T}_2 \models \text{ran}(r) \sqcap D \sqsubseteq X_i$, $(x_{X_i}, x_{Y'}, \varepsilon) \in \leftarrow^{\text{ran}}$, and X_i is non-conjunctive in \mathcal{T}_1 , we have that $\mathcal{T}_2 \models \text{ran}(r) \sqcap D \sqsubseteq Y'$ by applying one of the two previous cases. This implies that $\mathcal{T}_2 \models \text{ran}(r) \sqcap D \sqsubseteq Y'$ for every $Y_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$. Hence, $\mathcal{T}_2 \models \text{ran}(r) \sqcap D \sqsubseteq X'$. For b2), since $\mathcal{T}_2 \models D \sqsubseteq X_i$, $(x_{X_i}, x_{Y'}, \varepsilon) \in \leftarrow^{\text{ran}}$, and X_i is non-conjunctive in \mathcal{T}_1 we obtain $\mathcal{T}_2 \models D \sqsubseteq Y'$ by applying one of the previous two cases again. We have established that $\mathcal{T}_2 \models D \sqsubseteq Y'$ for every $Y_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$. Hence, $\mathcal{T}_2 \models D \sqsubseteq X_2$.

□

Lemma 185 *If $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{Ran}} \langle \mathcal{T}_2, X_2 \rangle$, then: for every $\mathcal{EL}_\Sigma^{\text{ran}}$ -concept D such that $\mathcal{T}_1 \models D \sqcap \text{ran}(\zeta) \sqsubseteq X$, it holds that $\mathcal{T}_2 \models D \sqcap \text{ran}(\zeta) \sqsubseteq X'$.*

Proof We show the lemma by induction on the structure of D . Let D be an $\mathcal{EL}_\Sigma^{\text{ran}}$ -concept and let $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{Ran}} \langle \mathcal{T}_2, X_2 \rangle$ with $X \in \mathbb{N}_C$ such that $\mathcal{T}_1 \models D \sqsubseteq X$ if $\zeta = \varepsilon$, or $\mathcal{T}_1 \models \text{ran}(r) \sqcap D \sqsubseteq X$ if $\zeta = r$. Note that $D \neq \top$ if $\zeta = \varepsilon$. We distinguish between the following cases, which are mutually exclusive as \mathcal{T}_1 is a terminology:

- X is pseudo-primitive in \mathcal{T}_1 : as $\text{ran}(\zeta) \sqcap D$ is an $\mathcal{EL}^{\text{ran}}$ -concept, it follows from Lemma 34 that there exists $Z \in \Sigma^{(\text{Ran}, \zeta)}$ such that $\mathcal{T}_1 \models Z \sqsubseteq X$.
As $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{Ran}} \langle \mathcal{T}_2, X_2 \rangle$ and $Z \in \Sigma^{(\text{Ran}, \zeta)}$, it follows from Item $(S_{\mathbb{N}_C}^{\leftarrow, \mathcal{L}})$ of Definition 160 that $\mathcal{T}_2 \models X \sqsubseteq Y_2$ for every $Y_2 \in \text{non-conj}_{\mathcal{T}_2}(X)$ with $\mathcal{T}_2 \not\models \text{ran}(\zeta) \sqsubseteq Y'$ if $\zeta \neq \varepsilon$. Hence, $\mathcal{T}_2 \models \text{ran}(\zeta) \sqcap D \sqsubseteq X'$. by Lemma 34.

- there exists an axiom of the form $X_1 \equiv \exists s.B_1$ in \mathcal{T}_1 : By Lemma 34 we have that either a1) there exists $Z \in \Sigma^{(\text{Ran}, \zeta)}$ such that $\mathcal{T}_1 \models Z \sqsubseteq X_1$, or a2) there exists a conjunct $\exists t.D'$ of D such that $\mathcal{T}_1 \models t \sqsubseteq s$ and $\mathcal{T}_1 \models \text{ran}(t) \cap D' \sqsubseteq B_1$. For a1), we proceed as above and obtain $\mathcal{T}_2 \models \text{ran}(\zeta) \cap D \sqsubseteq X$. Consider a2). As $\mathcal{T}_1 \models \text{ran}(t) \cap D' \sqsubseteq B$ and $\text{ran}(t) \cap D'$ is an $\mathcal{E}\mathcal{L}_\Sigma^{\text{ran}}$ -concept, it follows Definition 158 that B is (Σ, Ran, t) -entailed. Let $Y_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$. As $t \in \mathbb{N}_R \cap \Sigma$ with $\mathcal{T}_1 \models t \sqsubseteq s$, we obtain by Item $(S_{\exists}^{\leftarrow, \mathcal{L}})$ of Definition 160 that either i) $\mathcal{T}_2 \models \text{dom}(t) \sqsubseteq Y_2$, ii) $\mathcal{T}_2 \models \text{ran}(r) \sqsubseteq Y_2$ for $\zeta = r$, or iii) there exists $Y_2 \equiv \exists r'.B_2 \in \mathcal{T}_2$ with $\mathcal{T}_2 \models t \sqsubseteq r'$ and $\langle \mathcal{T}_1, B_1 \rangle \sim_{\Sigma, t}^{\leftarrow, \text{Ran}} \langle \mathcal{T}_2, B_2 \rangle$. For i), we have that $\mathcal{T}_2 \models \exists t.D' \sqsubseteq Y_2$. For ii), we obtain $\mathcal{T}_2 \models \text{ran}(r) \sqsubseteq Y'$. For iii), since $\mathcal{T}_1 \models \text{ran}(t) \cap D' \sqsubseteq B_1$, and $\text{ran}(t) \cap D'$ is an $\mathcal{E}\mathcal{L}_\Sigma^{\text{ran}}$ -concept, the induction hypothesis yields that $\mathcal{T}_2 \models \text{ran}(t) \cap D' \sqsubseteq B_2$. Then $\mathcal{T}_2 \models \exists t.(\text{ran}(t) \cap D') \sqsubseteq \exists t.B_2$ and, thus, $\mathcal{T}_2 \models \exists t.D' \sqsubseteq Y_2$. For $\zeta = \varepsilon$, we have that $\mathcal{T}_2 \models \exists t.D' \sqsubseteq Y_2$ for every $Y_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$. Hence, $\mathcal{T}_2 \models \exists t.D' \sqsubseteq X_2$. For $\zeta = r$, we obtain that $\mathcal{T}_2 \models \exists t.D' \sqsubseteq Y'$ or $\mathcal{T}_2 \models \text{ran}(r) \sqsubseteq Y_2$ for every $Y_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$. Hence, $\mathcal{T}_2 \models \text{ran}(r) \cap \exists t.D' \sqsubseteq X_2$.

- there exists an axiom of the form $X_1 \equiv X_{11} \cap \dots \cap X_{1n}$ in \mathcal{T}_1 :

Let $Y_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$. By Item $(S_{\cap}^{\leftarrow, \mathcal{L}})$ of Definition 160 we have that b1) for $\zeta = r$: $\mathcal{T}_2 \models \text{ran}(r) \sqsubseteq Y_2$ or there exists $i \in \{1, \dots, n\}$ such that $\mathcal{T}_1 \not\models \text{ran}(r) \sqsubseteq X_{1i}$ and $\langle \mathcal{T}_1, X_{1i} \rangle \sim_{\Sigma, \varepsilon}^{\leftarrow, \text{Ran}} \langle \mathcal{T}_2, Y_2 \rangle$, or b2) for $\zeta = \varepsilon$: $\langle \mathcal{T}_1, X_{1i} \rangle \sim_{\Sigma, \varepsilon}^{\leftarrow, \text{Ran}} \langle \mathcal{T}_2, Y_2 \rangle$ for some $i \in \{1, \dots, n\}$. For b1), we have that $\mathcal{T}_2 \models \text{ran}(r) \sqsubseteq Y_2$, or, since $\mathcal{T}_2 \models \text{ran}(r) \cap D \sqsubseteq X_i$, $\langle \mathcal{T}_1, X_{1i} \rangle \sim_{\Sigma, \varepsilon}^{\leftarrow, \text{Ran}} \langle \mathcal{T}_2, Y_2 \rangle$, and X_{1i} is non-conjunctive in \mathcal{T}_1 we have that $\mathcal{T}_2 \models \text{ran}(r) \cap D \sqsubseteq Y_2$ by applying one of the two previous cases. This implies that $\mathcal{T}_2 \models \text{ran}(r) \cap D \sqsubseteq Y_2$ for every $Y_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$. Hence, $\mathcal{T}_2 \models \text{ran}(r) \cap D \sqsubseteq X_2$. For b2), since $\mathcal{T}_2 \models D \sqsubseteq X_{1i}$, $\langle \mathcal{T}_1, X_{1i} \rangle \sim_{\Sigma, \varepsilon}^{\leftarrow, \text{Ran}} \langle \mathcal{T}_2, Y_2 \rangle$, and X_i is non-conjunctive in \mathcal{T}_1 , we obtain $\mathcal{T}_2 \models D \sqsubseteq Y_2$ by applying one of the previous two cases again. We have established that $\mathcal{T}_2 \models D \sqsubseteq Y_2$ for every $Y_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$. Hence, $\mathcal{T}_2 \models D \sqsubseteq X_2$. □

Lemma 185 *If $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{Ran}} \langle \mathcal{T}_2, X_2 \rangle$, then: for every $\mathcal{E}\mathcal{L}_\Sigma^{\text{ran}}$ -concept D such that $\mathcal{T}_1 \models D \cap \text{ran}(\zeta) \sqsubseteq X$, it holds that $\mathcal{T}_2 \models D \cap \text{ran}(\zeta) \sqsubseteq X'$.*

Proof The statement of the lemma follows from Lemma 184 and Lemma 185. □

We then give the definition of $\text{sub}(D)$ for $\mathcal{E}\mathcal{L}^{\text{ran}}$ -concept D , which will be used in the following proof.

Definition 186 *Let D be an $\mathcal{E}\mathcal{L}^{\text{ran}}$ -concept, $A \in \mathbb{N}_C$ and $r \in \mathbb{N}_R$.*

We define the set $\text{sub}(D) \subseteq \mathcal{E}\mathcal{L}^{\text{ran}}$ inductively as follows:

- $\text{sub}(\top) = \{\top\}$;
- $\text{sub}(A) = \{A\}$;
- $\text{sub}(\text{dom}(r)) = \{\text{dom}(r)\}$;

- $\text{sub}(\text{ran}(r)) = \{\text{ran}(r)\};$
- $\text{sub}(\exists r.D) = \{\exists r.D\} \cup \text{sub}(D);$
- $\text{sub}(D_1 \sqcap D_2) = \{D_1 \sqcap D_2\} \cup \text{sub}(D_1) \cup \text{sub}(D_2).$

Moreover, we define the set $\text{sub}_0(C) \subseteq \mathcal{ELH}^r$ inductively as follows

- $\text{sub}_0(\top) = \{\top\};$
- $\text{sub}_0(A) = \{A\};$
- $\text{sub}_0(\text{dom}(r)) = \{\text{dom}(r)\};$
- $\text{sub}_0(\text{ran}(r)) = \{\text{ran}(r)\};$
- $\text{sub}_0(\exists r.C) = \{\exists r.C\};$
- $\text{sub}_0(C_1 \sqcap C_2) = \text{sub}_0(C_1) \cup \text{sub}_0(C_2).$

Lemma 187 *Let \mathcal{T} be a normalised and acyclic \mathcal{ELH}^r -terminology, let Σ be a signature, and let $X \in \mathbb{N}_C$ be pseudo-primitive in \mathcal{T} .*

Then X is not complex $\langle \Sigma, \mathcal{L}, \zeta \rangle$ -entailed w.r.t. \mathcal{T} .

Proof As X is pseudo-primitive in \mathcal{T} , we have $\text{non-conj}_{\mathcal{T}}(X) = \{X\}$. We can infer that there cannot exist $B \in \Sigma$ with $\mathcal{T} \models B \sqsubseteq Y$ and $\mathcal{T} \not\models B \sqsubseteq X$. Additionally, we observe that no axiom of the form $X \equiv \exists r.Y$ is contained in \mathcal{T} , which implies that X is not complex $\langle \Sigma, \mathcal{L}, \zeta \rangle$ w.r.t. \mathcal{T} . \square

Lemma 188 *Let \mathcal{T} be a normalised and acyclic \mathcal{ELH}^r -terminology and let $X \in \mathbb{N}_C$ is not complex $\langle \Sigma, \text{Ran}, \zeta \rangle$ -entailed w.r.t. \mathcal{T} .*

Then for every $D \in \mathcal{EL}_{\Sigma}^{\text{ran}}$ it holds that: $\mathcal{T} \models D \sqsubseteq X$ iff there exists $B \in \text{sub}_0(D)$ such that $\mathcal{T} \models B \sqsubseteq X$.

Proof Let $D \in \mathcal{EL}_{\Sigma}^{\text{ran}}$.

“only if”: We assume that $\mathcal{T} \models D \sqsubseteq X$. Note that $\mathcal{T} \not\models \top \sqsubseteq D$ as \mathcal{T} is a normalised terminology, which implies that $\top \notin \text{sub}(\mathcal{T})$. Then, as X is not complex Σ -entailed w.r.t. \mathcal{T} , it follows that there exists $Y \in \text{non-conj}_{\mathcal{T}}(X)$ such that

- (i) for every $B \in \Sigma^{(\text{Ran}, \zeta)}$ with $\mathcal{T} \models B \sqsubseteq Y$ it holds that $\mathcal{T} \models B \sqsubseteq X$, and
- (ii) for every $Y \equiv \exists s.Y' \in \mathcal{T}$, it holds that either s is not Σ -entailed or Y' is not $\langle \Sigma, \text{Ran}, \zeta \rangle$ -entailed in \mathcal{T} .

Note that $\mathcal{T} \models D \sqsubseteq Y$ since $\mathcal{T} \models X \sqsubseteq Y$. We now distinguish between the following cases. If Y is pseudo-primitive in \mathcal{T} , then it follows from Lemma 34 that there exists $B \in \text{sub}_0(D)$, $B \in \mathbb{N}_C \cup \{\text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_R\}$ with $\mathcal{T} \models B \sqsubseteq Y$, which implies that $\mathcal{T} \models B \sqsubseteq X$ by (i).

Otherwise, there exists an axiom $Y \equiv \exists s.Y' \in \mathcal{T}$ and it holds that $\mathcal{T} \models D \sqsubseteq \exists s.Y'$. It follows from Lemma 35 that either (a) there exists $B \in \text{sub}_0(C)$, $B \in \text{sub}_0(D)$, $B \in \mathbb{N}_C \cup \{\text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_R\}$ with $\mathcal{T} \models B \sqsubseteq Y$, or (b) there exists $D' \sqcap \text{ran}(s') \in \text{sub}(D)$ with $\mathcal{T} \models D' \sqcap \text{ran}(s') \sqsubseteq Y'$ and $\mathcal{T} \models s' \sqsubseteq s$. For (a), we have $\mathcal{T} \models B \sqsubseteq Y$, and thus, $\mathcal{T} \models B \sqsubseteq X$ by (i). We note that (b) cannot hold as $\text{sig}(D') \subseteq \Sigma$, which implies that $\mathcal{T} \models D' \sqcap \text{ran}(s) \sqsubseteq Y'$ always holds, then Y' is $\langle \Sigma, \text{Ran}, \zeta \rangle$ -entailed, which contradicts to (ii). So $\mathcal{T} \models B \sqsubseteq X$.

“if”: Let $B \in \text{sub}_0(D)$ such that $\mathcal{T} \models B \sqsubseteq X$. Since $\mathcal{T} \models D \sqsubseteq B$, it follows that $\mathcal{T} \models D \sqsubseteq X$. \square

Lemma 189 *Let \mathcal{T} be a normalised and acyclic $\mathcal{EL}\mathcal{H}'$ -terminology and let $X \in \mathbb{N}_C$ is not complex $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed w.r.t. \mathcal{T} . Then for every $C \in \mathcal{EL}_\Sigma$ it holds that: $\mathcal{T} \models C \sqcap \text{ran}(\zeta) \sqsubseteq X$ iff there exists $B \in \text{sub}_0(C)$ such that $\mathcal{T} \models B \sqsubseteq X$.*

Proof Let $C \in \mathcal{EL}_\Sigma$.

“only if”: We assume that $\mathcal{T} \models C \sqcap \text{ran}(\zeta) \sqsubseteq X$. Note that $\mathcal{T} \not\models \top \sqsubseteq C$ as \mathcal{T} is a normalised terminology, which implies that $\top \notin \text{sub}(\mathcal{T})$. Then, as X is not complex Σ -entailed w.r.t. \mathcal{T} , it follows that there exists $Y \in \text{non-conj}_{\mathcal{T}}(X)$ such that

- (i) for every $B \in \Sigma^{\langle \text{ran}, \zeta \rangle}$ with $\mathcal{T} \models B \sqsubseteq Y$ it holds that $\mathcal{T} \models B \sqsubseteq X$, and
- (ii) for every $Y \equiv \exists s.Y' \in \mathcal{T}$, it holds that either s is not Σ -entailed or Y' is not $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed in \mathcal{T} .

Note that $\mathcal{T} \models C \sqcap \text{ran}(\zeta) \sqsubseteq Y$ since $\mathcal{T} \models X \sqsubseteq Y$. We now distinguish between the following cases:

If Y is pseudo-primitive in \mathcal{T} , then it follows from Lemma 34 that there exists $B \in \text{sub}_0(C)$, $B \in \mathbb{N}_C \cup \{\text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_R\}$ with $\mathcal{T} \models B \sqsubseteq Y$, which implies that $\mathcal{T} \models B \sqsubseteq X$ by (i).

Otherwise, there exists an axiom $Y \equiv \exists s.Y' \in \mathcal{T}$ and it holds that $\mathcal{T} \models C \sqcap \text{ran}(\zeta) \sqsubseteq \exists s.Y'$. It follows from Lemma 35 that either (a) there exists $B \in \text{sub}_0(C)$, $B \in \mathbb{N}_C \cup \{\text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_R\}$ with $\mathcal{T} \models B \sqsubseteq Y$, or (b) there exists $C' \sqcap \text{ran}(s') \in \text{sub}(C)$ with $\mathcal{T} \models C' \sqcap \text{ran}(s') \sqsubseteq Y'$ and $\mathcal{T} \models s' \sqsubseteq s$. For (a), we have $\mathcal{T} \models B \sqsubseteq Y$, and thus, $\mathcal{T} \models B \sqsubseteq X$ by (i). We note that (b) cannot hold as $\text{sig}(C') \subseteq \Sigma$, which implies that $\mathcal{T} \models C' \sqcap \text{ran}(s') \sqsubseteq Y'$ always holds, then Y' is $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed, which contradict to (ii). So $\mathcal{T} \models B \sqsubseteq X$.

“if”: let $B \in \text{sub}_0(C)$ such that $\mathcal{T} \models B \sqsubseteq X$. Since $\mathcal{T} \models C \sqcap \text{ran}(\zeta) \sqsubseteq B$, it follows that $\mathcal{T} \models C \sqcap \text{ran}(\zeta) \sqsubseteq X$. \square

Lemma 190 *Let $\chi = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran} \rangle$ and $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \text{COVER}_{\leftarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran}, \zeta)$. If $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2, X_2 \rangle$, then $\mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ is the set such that for every $\mathcal{M} \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$, $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{M}, X_2 \rangle$ and for any $\mathcal{M}' \subseteq \mathcal{M}$, $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{M}', X_2 \rangle$ does not hold.*

Proof First we prove that $\mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ is a cover of the set \mathcal{M} such that for every $\mathcal{M} \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$, $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{M}, X_2 \rangle$ hold.

(i) We show that every $\mathcal{M} \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$, $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{M}, X_2 \rangle$ holds. That is, we prove that for every $C \in \mathcal{E}\mathcal{L}_\Sigma$, $\mathcal{T}_1 \models \text{ran}(\zeta) \cap C \sqsubseteq X_1$ implies that $\mathcal{M} \models \text{ran}(\zeta) \cap C \sqsubseteq X_2$ and $\mathcal{M} \subseteq \mathcal{T}_2$.

First, we define the relation $\succ_{\mathcal{T}} \subseteq \mathbb{N}_C \times \mathbb{N}_C$ as follows: for $X, Y \in \mathbb{N}_C$ we set $X \succ Y$ iff either there exists $X \equiv \exists r.Y \in \mathcal{T}$ or there exists $X \equiv Y \cap \dots \cap Y_n \in \mathcal{T}$ such that $Y_i = Y$ for some $1 \leq i \leq n$. Note that $\succ_{\mathcal{T}}$ is well-founded as \mathcal{T} is acyclic. The proof now proceeds by induction on $\succ_{\mathcal{T}_2}$.

Let $X_1, X_2 \in \mathbb{N}_C$ such that $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2, X_2 \rangle$, and let $\mathcal{M} \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$. We now distinguish between the following cases.

If X_1 is not $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed in \mathcal{T}_1 , then $\mathcal{M} = \emptyset$ and there does not exist $C \in \mathcal{E}\mathcal{L}_\Sigma$ with $\mathcal{T}_1 \models \text{ran}(\zeta) \cap C \sqsubseteq X_1$. Hence, it trivially holds for every $C \in \mathcal{E}\mathcal{L}_\Sigma$ with $\mathcal{T}_1 \models \text{ran}(\zeta) \cap C \sqsubseteq X_1$ that $\mathcal{M} \models \text{ran}(\zeta) \cap C \sqsubseteq X_2$.

Otherwise, X_1 is $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed w.r.t. \mathcal{T}_1 . We further distinguish between the following cases. If X_1 is not complex $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed w.r.t. \mathcal{T}_1 , then let $C \in \mathcal{E}\mathcal{L}_\Sigma$ such that $\mathcal{T}_1 \models C \sqsubseteq X_1$. Then by Lemma 189 there exists $B \in \text{sub}_0(C)$ such that $\mathcal{T}_1 \models B \sqsubseteq X_1$. As $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2, X_2 \rangle$, there exists $J \in \text{Just}_{\mathcal{T}_2}(B \sqsubseteq X_2)$ with $J \subseteq \mathcal{M}$, which implies that $\mathcal{M} \models C \sqsubseteq X_2$, hence, $\mathcal{M} \models \text{ran}(\zeta) \cap C \sqsubseteq X_2$.

Otherwise, X_1 is complex $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed w.r.t. \mathcal{T}_1 . We distinguish between the following cases:

- If $X_1 \equiv \exists r.Y_1 \in \mathcal{T}_1$ such that r is not Σ -entailed or Y_1 is not $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed w.r.t. \mathcal{T}_1 . then let $C \in \mathcal{E}\mathcal{L}_\Sigma$ such that $\mathcal{T}_1 \models \text{ran}(\zeta) \cap C \sqsubseteq X_1$, and therefore $\mathcal{T}_1 \models \text{ran}(\zeta) \cap C \sqsubseteq \exists r.Y_1$. By Lemma 34 we can infer that there exists $B \in \text{sub}_0(C)$ ($B \in \mathbb{N}_C \cup \{\text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_R\}$) with $\mathcal{T}_1 \models B \sqsubseteq \exists r.Y_1$. We can then show that $\mathcal{M} \models C \sqsubseteq X_2$ as above.
- If $X_1 \equiv \exists r.Y_1 \in \mathcal{T}_1$ such that $\mathcal{T}_1 \models s \sqsubseteq r$, $s \in \Sigma$ and Y_1 is $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed w.r.t. \mathcal{T}_1 , then let $C \in \mathcal{E}\mathcal{L}_\Sigma$ such that $\mathcal{T}_1 \models C \cap \text{ran}(\zeta) \sqsubseteq X_1$, and therefore $\mathcal{T}_1 \models C \cap \text{ran}(\zeta) \sqsubseteq \exists r.Y_1$. By Lemma 34 either (a) there exists $B \in \text{sub}_0(C)$ ($B \in \mathbb{N}_C \cup \{\text{dom}(r), \text{ran}(r) \mid r \in \mathbb{N}_R\}$) with $\mathcal{T}_1 \models B \sqsubseteq \exists r.Y_1$, or (b) there exists $\exists r'.C' \in \text{sub}_0(C)$ with $\mathcal{T}_1 \models C' \cap \text{ran}(r') \sqsubseteq Y_1$ and $\mathcal{T}_1 \models r' \sqsubseteq s$.

For (a), we can show that $\mathcal{M} \models C \cap \text{ran}(\zeta) \sqsubseteq X_2$ as above.

For (b), as $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2, X_2 \rangle$, we can infer that for every $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$, there exists $X'_2 \equiv \exists r'.Y'_2 \in \mathcal{T}_2$ such that $\mathcal{T}_2 \models r' \sqsubseteq s$ and $\langle \mathcal{T}_1, Y_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2, Y'_2 \rangle$. Moreover, for every $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$ not entailed by $\text{dom}(s)$ or $\text{ran}(\zeta)$, there exists $\mathcal{M}'_{Y'_2} \in \mathbb{M}_{(Y_1, Y'_2)}^{\leftarrow}$ (Line 6 in Algorithm 24) such that $\mathcal{M}'_{Y'_2} \cup \{X'_2 \equiv \exists r.Y'_2\} \subseteq \mathcal{M}$ (Line 8 in Algorithm 24).

By applying the induction hypothesis we have that for every $\mathcal{M}'_{Y'_2} \in \mathbb{M}_{(Y_1, Y'_2)}^{\leftarrow}$, $\langle \mathcal{T}_1, Y_1 \rangle \sim_{\Sigma, s}^{\leftarrow, \text{ran}} \langle \mathcal{M}', Y'_2 \rangle$ holds.

Consequently, for every $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$ that does not entailed by $\text{dom}(s)$ or $\text{ran}(\zeta)$, we have that $\mathcal{M} \models C' \cap \text{ran}(s) \sqsubseteq Y'_2$, which implies $\mathcal{M} \models \exists r.(C' \cap \text{ran}(s)) \sqsubseteq \exists r.Y'_2 \equiv X'_2$ then we infer that $\mathcal{M} \models \exists r.(C' \cap \text{ran}(s)) \sqsubseteq \exists r.Y'_2 \equiv X'_2$ for every $X'_2 \in \text{non-conj}_{\mathcal{T}_2}(X_2)$ that does

not entailed by $\text{dom}(s)$ or $\text{ran}(\zeta)$. As $\exists r'. C' \in \text{sub}_0(C)$ and $\text{max-tree}_{\mathcal{T}_2}^\square(X_2) \subseteq \mathcal{M}$ (Line 3 in Algorithm 24), we have $\mathcal{M} \models \text{ran}(\zeta) \cap C \sqsubseteq X_2$.

If $X_1 \equiv Y_1 \sqcap \dots \sqcap Y_m \in \mathcal{T}_1$, then let $C \in \mathcal{EL}_\Sigma$ such that $\mathcal{T}_1 \models \text{ran}(\zeta) \cap C \sqsubseteq X_1$, i.e. $\mathcal{T}_1 \models \text{ran}(\zeta) \cap C \sqsubseteq X'_1$ for every $X'_1 \in \text{non-conj}_{\mathcal{T}_1}(X_1)$ such that $\mathcal{T}_2 \models \text{ran}(r) \sqsubseteq X'_2$. As $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} \neq \emptyset$, we can infer that there exists $\Gamma \in \text{DefForest}_{\mathcal{T}_2}^\square(X_2)$ such that for every $X'_2 \in \text{leaves}(\Gamma)$, there exists $\psi \in \text{non-conj}_{\mathcal{T}_1}(X_1)$ such that $\langle \mathcal{T}_1, \psi \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2 \setminus \delta_\Gamma, X'_2 \rangle$. We can get that there exists $\mathcal{M}'_{X'_2} \in \text{COVER}_{\leftarrow}(\mathcal{T}_1, \mathcal{T}_2 \setminus \delta_\Gamma, \Sigma, \psi, X'_2, \text{ran}, \varepsilon)$ with $\mathcal{M}'_{X'_2} \subseteq \mathcal{M}$ (Lines 4 to 17 in Algorithm 25). By applying the induction hypothesis we have for every $X'_2 \in \text{leaves}(\Gamma)$, $\langle \mathcal{T}_1, \psi \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{M}'_{X'_2}, X'_2 \rangle$ where $\mathcal{M}'_{X'_2} \subseteq \mathcal{T}_2 \setminus \delta_\Gamma$ i.e. $\mathcal{M}'_{X'_2} \models C \sqsubseteq X'_2$ (as $\zeta = \varepsilon$) and $\mathcal{M} \models C \sqsubseteq X'_2$ hold as $\mathcal{T}_1 \models C \sqsubseteq \psi$. Consequently, we have $\mathcal{M} \models C \sqsubseteq X_2$ as $\Gamma \subseteq \mathcal{M}$ (Line 14 in Algorithm 25).

(ii) We show that for every $X_1, X_2 \in \mathbb{N}_C$ with $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2, X_2 \rangle$, and for every subsumee module \mathcal{M} such that $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2, X_2 \rangle$, there exists $\mathcal{M}' \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ such that $\mathcal{M}' \subseteq \mathcal{M}$.

First, we define the relation $\succ_{\mathcal{T}_1} \subseteq \mathbb{N}_C \times \mathbb{N}_C$ as follows: for $X, Y \in \mathbb{N}_C$ we set $X \succ Y$ iff either there exists $X \equiv \exists r. Y \in \mathcal{T}_1$ or there exists $X \equiv Y_1 \sqcap \dots \sqcap Y_n \in \mathcal{T}_1$ such that $Y_i = Y$ for some $1 \leq i \leq n$. Note that \succ is well-founded as \mathcal{T}_1 is acyclic.

The proof now proceeds by induction on \succ . Let $X_1, X_2 \in \mathbb{N}_C$ with $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2, X_2 \rangle$ and let \mathcal{M} be a subsumee module under $\chi = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran}, \zeta \rangle$. Additionally, let S be a Σ -subsumee simulation from \mathcal{T}_1 to \mathcal{M} with $(X_1, X_2, \zeta) \in S$.

First, we observe that for every $\phi \in \Sigma^{(\text{ran}, \zeta)}$ with $\mathcal{T}_1 \models B \sqsubseteq X_1$, there exists $J_{\phi \sqsubseteq X_1}^{\mathcal{M}} \in \text{Just}_{\mathcal{T}_2}(\phi \sqsubseteq X_2)$ with $J_{\phi \sqsubseteq X_1}^{\mathcal{M}} \subseteq \mathcal{M}$ as \mathcal{M} is a subsumee module under $\chi = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran}, \zeta \rangle$.

We now distinguish between the following cases. If X_1 is not $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed w.r.t. \mathcal{T}_1 , we set $\mathcal{M}' := \emptyset$. Then it holds that $\mathcal{M}' \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ (Line 3 in Algorithm 26) and $\mathcal{M}' \subseteq \mathcal{M}$.

Otherwise, X_1 is $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed w.r.t. \mathcal{T}_1 , and we proceed as follows. If X_1 is not complex $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed w.r.t. \mathcal{T}_1 , we set $\mathcal{M}' := \bigcup_{\phi \in \Sigma^{\text{ran}, \zeta}} J_{\phi \sqsubseteq X_1}^{\mathcal{M}}$. Then it holds that $\mathcal{M}' \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ (Line 5 in Algorithm 26 and Algorithm 23) and $\mathcal{M}' \subseteq \mathcal{M}$.

Otherwise, X_1 is complex $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed w.r.t. \mathcal{T}_1 , which implies that X_1 is not pseudo-primitive in \mathcal{T}_1 by Lemma 187. We now distinguish between the following two cases:

1. If $X_1 \equiv \exists r. Y_1 \in \mathcal{T}_1$, we analyse the following subcases:

- If $r \notin \Sigma$ or Y_1 is not $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed w.r.t. \mathcal{T}_1 , then for $\mathcal{M}' := \bigcup_{\phi \in \Sigma} J_{\phi \sqsubseteq X_1}^{\mathcal{M}}$ it holds that $\mathcal{M}' \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ (Line 5 in Algorithm 26 and Algorithm 23) and $\mathcal{M}' \subseteq \mathcal{M}$.
- Otherwise, we assume that Y_1 and r are $\langle \Sigma, \text{ran}, \zeta \rangle$ -entailed w.r.t. \mathcal{T}_1 , $\mathcal{T} \models s \sqsubseteq r$ and $s \in \Sigma$. By Condition $(S_{\exists}^{\rightarrow})$ we have that for every $X'_2 \in \text{non-conj}_{\mathcal{M}}(X_2)$ not entailed by $\text{dom}(s)$ or $\text{ran}(\zeta)$ w.r.t. \mathcal{T}_2 there exists $X'_2 \equiv \exists r. Y'_2 \in \mathcal{M}$ such that $(Y_1, Y'_2, s) \in S$.

Now we show that $\text{max-tree}_{\mathcal{T}_2}^\square(X_2) \subseteq \mathcal{M}$ by showing that

$$\text{max-tree}_{\mathcal{T}_2}^\square(X_2) = \text{max-tree}_{\mathcal{M}}^\square(X_2).$$

It immediately holds that $\text{max-tree}_{\mathcal{M}}^{\square}(X_2) \subseteq \text{max-tree}_{\mathcal{T}_2}^{\square}(X_2)$ as $\mathcal{M} \subseteq \mathcal{T}_2$. Let $\alpha \in \text{max-tree}_{\mathcal{T}_2}^{\square}(X_2)$. If we assume towards a contradiction that $\alpha \notin \text{max-tree}_{\mathcal{M}}^{\square}(X_2)$, then there exists $\xi \in \text{leaves}(\text{max-tree}_{\mathcal{M}}^{\square}(X_2))$ such that $\xi \in \text{def}_{\mathcal{T}_2}^{\square}$ and $\xi \equiv \exists r. \psi \in \mathcal{M} \subseteq \mathcal{T}_2$. We have derived a contradiction.

Let $X'_2 \in \text{non-conj}_{\mathcal{M}}(X_2)$ such that X'_2 not entailed by $\text{dom}(s)$ or $\text{ran}(\zeta)$. By Condition $(S_{\exists}^{\rightarrow})$ there exists $\alpha_{X'_2} := X'_2 \equiv \exists r'. Y'_2 \in \mathcal{M}$ with $\mathcal{T}_2 \models s \sqsubseteq r'$, $s \in \Sigma$ and $(Y_1, Y'_2, s) \in S$. Consequently, it holds that $\langle \mathcal{T}_1, Y_1 \rangle \sim_{\Sigma, s}^{\leftarrow, \text{ran}} \langle \mathcal{M}, X_2 \rangle$. By applying the induction hypothesis there exists $\mathcal{M}'_{X'_2} \in \mathbb{M}_{(Y_1, Y'_2)}^{\leftarrow}$ such that $\mathcal{M}'_{X'_2} \subseteq \mathcal{M}$.

We set

$$\begin{aligned} \mathcal{M}' := & \bigcup_{\substack{\phi \in \Sigma^{\text{ran}, \zeta} \\ \mathcal{T}_2 \models \phi \sqsubseteq X_2}} J_{\phi \sqsubseteq X_2}^{\mathcal{M}} \cup \{\text{max-tree}_{\mathcal{T}_2}^{\square}(X_2)\} \cup \text{Just}_{\mathcal{T}_2}(s \sqsubseteq r') \\ & \cup \bigcup_{\substack{X'_2 \in \text{non-conj}_{\mathcal{M}}(X_2) \\ \mathcal{T}_2 \models \text{ran}(\zeta) \sqsubseteq X'_2}} \mathcal{M}'_{X'_2} \cup \{\alpha_{X'_2}\}. \end{aligned}$$

Then it holds that $\mathcal{M}' \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ (Line 10 in Algorithm 26 and Line 7 in Algorithm 24) and $\mathcal{M}' \subseteq \mathcal{M}$.

2. If $X_1 \equiv Y_1 \sqcap \dots \sqcap Y_m \in \mathcal{T}_1$, then for every $X'_2 \in \text{non-conj}_{\mathcal{M}}(X_2)$ that is not entailed by $\text{ran}(\zeta)$ w.r.t. \mathcal{T}_2 , there exists $\psi \in \text{non-conj}_{\mathcal{T}_1}(X_1)$ such that $\langle \mathcal{T}_1, \psi \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2 \setminus \delta_{\Gamma}, X'_2 \rangle$ by Condition $(S_{\sqcap}^{\leftarrow, \mathcal{L}})$. Hence, for $\Gamma := \text{max-tree}_{\mathcal{M}}^{\square}(X_2)$ it holds that $\Gamma \in \text{DefForest}_{\mathcal{T}_2}^{\square}(X_2)$ and $\Gamma \subseteq \mathcal{M}$. Moreover, for every $X'_2 \in \text{non-conj}_{\mathcal{M}}(X_2)$ that is not entailed by $\text{ran}(\zeta)$ w.r.t. \mathcal{T}_2 , it holds that \mathcal{M} is a subsumee module under $\chi_{\psi} = \langle \mathcal{T}_1, \psi, \Sigma, \mathcal{T}_2, X'_2, \text{ran}, \zeta \rangle$. By applying the induction hypothesis for every $X'_2 \in \text{non-conj}_{\mathcal{M}}(X_2)$, there exists $\mathcal{M}'_{X'_2} \in \mathbb{M}_{(\psi, X'_2)}^{\leftarrow}$ such that $\mathcal{M}'_{X'_2} \subseteq \mathcal{M}$.

We set

$$\mathcal{M}' := \bigcup_{\substack{\phi \in \Sigma \\ \mathcal{T}_2 \models \phi \sqsubseteq X_2}} J_{\phi \sqsubseteq X_2}^{\mathcal{M}} \cup \Gamma \cup \bigcup_{\substack{X'_2 \in \text{non-conj}_{\mathcal{M}}(X_2) \\ \mathcal{T}_2 \models \text{ran}(\zeta) \sqsubseteq X'_2}} \mathcal{M}'_{X'_2}.$$

Note that for every $X'_2 \in \text{non-conj}_{\mathcal{M}}(X_2)$ that is not entailed by $\text{ran}(\zeta)$ w.r.t. \mathcal{T}_2 , it holds that $\langle \mathcal{T}_1, \phi \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2 \setminus \delta_{\Gamma}, X'_2 \rangle$, as $(\phi, X'_2, \varepsilon) \in S$ and $\mathcal{M} \subseteq \mathcal{T}_2 \setminus \delta_{\Gamma}$.

Now we have that for every $M \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$, $M, \langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2, X_2 \rangle$ holds by Definition 58.

We prove by assuming towards a contradiction. We assume now towards a contradiction that there exists an $\mathcal{M}' \subseteq \mathcal{M}$, that $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{M}', X_2 \rangle$ does not hold. Let $\alpha \in \mathcal{M} \setminus \mathcal{M}'$. As $\mathbb{M}_X^{\leftarrow}$ is initialed as $\{\emptyset\}$, we enumerate all possible situations that \mathcal{M} is generated in the algorithms But none of them is possible, as the absence of α will break the condition of subsumee simulation. Then $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{M}', X_2 \rangle$ does not hold anymore. Finally, we have that $\mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ is a cover of the set such that for every $\mathcal{M} \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$, $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{M}, X_2 \rangle$. \square

Theorem 163 Let $\chi = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \mathcal{L} \rangle$, where \mathcal{T}_1 and \mathcal{T}_2 are two acyclic and normalised $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies, Σ is a signature and $X_1, X_2 \in \Sigma \cap \mathbf{N}_C$. Additionally, let

$$\mathbb{M} := \text{COVER}_{\leftarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran}, \zeta).$$

If $\mathcal{T}_1 \sim_{\Sigma}^{\leftarrow} \mathcal{T}_2$, then for every $\mathcal{M} \in \mathbb{M}$, it holds that $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \mathcal{L}} \langle \mathcal{M}, X_2 \rangle$ and \mathcal{M} is minimal w.r.t. set inclusion.

Proof It follows from Lemma 190. □

Lemma 191 Let $\chi = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran} \rangle$ and

$$\mathbb{M}_{\text{ran}} := \text{COVER}_{\leftarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran}, \varepsilon) \otimes \bigotimes_{r \in \Sigma \cap \mathbf{N}_R} \text{COVER}_{\leftarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran}, r).$$

If $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{ran}} \langle \mathcal{T}_2, X_2 \rangle$, then $\mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ is the set of all subsumee projection justifications under χ .

Proof By Lemma 190 and Lemma 72, we have that for every $\mathcal{M} \in \mathbb{M}_{\text{ran}}$, \mathcal{M} is the minimal set such that $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \varepsilon}^{\leftarrow, \text{ran}} \langle \mathcal{M}, X_2 \rangle$, and $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, r}^{\leftarrow, \text{ran}} \langle \mathcal{M}, X_2 \rangle$. By Theorem 162, we have that for every $C, C' \in \mathcal{E}\mathcal{L}_{\Sigma}^{\text{ran}}$ with $\mathcal{T}_1 \models \text{ran}(r) \cap C \sqsubseteq X_1$ and $\mathcal{T}_1 \models C' \sqsubseteq X_1$, it holds that $\mathcal{T}_2 \models \text{ran}(r) \cap C \sqsubseteq X_2$ and $\mathcal{T}_2 \models C' \sqsubseteq X_2$. By Definition 137, we have that $\mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ is the set of all subsumee projection justifications under χ . □

Lemma 192 Let $\chi = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{Ran} \rangle$ and $\mathbb{M}_{(X_1, X_2)}^{\leftarrow} := \text{COVER}_{\leftarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{Ran}, \zeta)$. If $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{Ran}} \langle \mathcal{T}_2, X_2 \rangle$, then $\mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ is the set of all subsumee projection justifications under χ .

Proof Similar as proof of Lemma 190, we can proof that $\mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ is the set such that for $\mathcal{M} \in \mathbb{M}_{(X_1, X_2)}^{\leftarrow}$, $\langle \mathcal{T}_1, X_1 \rangle \sim_{\Sigma, \zeta}^{\leftarrow, \text{Ran}} \langle \mathcal{M}, X_2 \rangle$ and \mathcal{M} is minimal w.r.t. set inclusion. By Theorem 162, we have that for every $C \in \mathcal{E}\mathcal{L}_{\Sigma}^{\text{ran}}$ with $\mathcal{T}_1 \models C \sqsubseteq X_1$, it holds that $\mathcal{T}_2 \models C \sqsubseteq X_2$. By Definition 137, we have that $\mathbb{M}_{(X_1, X_2)}^{\leftarrow}$ is the set of all subsumee projection justifications under χ . □

Theorem 164 Let $\chi_{\text{ran}} = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran} \rangle$, $\chi_{\text{Ran}} = \langle \mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{Ran} \rangle$, where \mathcal{T}_1 and \mathcal{T}_2 are two acyclic and normalised $\mathcal{E}\mathcal{L}\mathcal{H}^r$ -terminologies, Σ is a signature and $X_1, X_2 \in \Sigma \cap \mathbf{N}_C$. Additionally, let

$$\mathbb{M}_{\text{ran}} := \text{Minimise}_{\subseteq}(\text{COVER}_{\leftarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran}, \varepsilon) \otimes \bigotimes_{r \in \Sigma \cap \mathbf{N}_R} \text{COVER}_{\leftarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran}, r)),$$

$$\mathbb{M}_{\text{Ran}} := \text{COVER}_{\leftarrow}(\mathcal{T}_1, X_1, \Sigma, \mathcal{T}_2, X_2, \text{ran}, \varepsilon).$$

If $\mathcal{T}_1 \sim_{\Sigma}^{\leftarrow} \mathcal{T}_2$, then \mathbb{M}_{ran} is the set of all subsumee justifications under χ_{Ran} and \mathbb{M}_{Ran} is the set of all subsumee justifications under χ_{Ran} .

Proof It follows from Lemma 191 and Lemma 192. \square

6.5 Evaluation

We implemented the algorithms introduced above in Java. Now we present a first evaluation of the proposed projection module extraction techniques on real-world biomedical ontologies with the help of PULi [22], a state-of-the-art tool for computing justifications for \mathcal{EL} ontologies. All experiments were conducted on machines equipped with an Intel Xeon Core 4 Duo CPU running at 2.50 GHz and with 64 GiB of RAM.

We consider computing projection modules between two prominent biomedical ontologies: MeSH¹ and Snomed CT². As MeSH is a hierarchically-organized terminology for indexing and cataloging of biomedical information. Therefore, MeSH was used as reference ontology in our experiment. We chose Snomed CT as an implementing ontology since it contains overwhelmingly many axioms. Automated tool support is needed for extracting the most relevant information to be shown to users. There exist 29962 common concept names between MeSH and Snomed CT. For every common concept name, we generated a signature by including its super-concepts that are also common concept names between these two terminologies. In total, 6572 signatures were generated this way. The number of concept names contained in each signature ranges from 2 to 13. The computation of projection modules for each signature finished within 1 second.

Note that when Snomed CT was constructed, MESH was not used as a guideline or a reference ontology. However, our experiment shows that for 19.8% of the signatures among those tested, there exist projection modules between MESH and Snomed CT, which illustrates the overlap between these two ontologies. The resulting projection modules are relatively small, thus lending themselves to manual inspection. Table 6.1 shows the minimal, maximal and median value of $\perp\top^*$ -module and projection module w.r.t. the generated signatures. One can see that, in general, projection module is smaller than the star modules of the implementing ontology SNOMED, so that users can focus on a smaller number of axioms in Snomed CT to verify how the knowledge of MESH about a signature is implemented in Snomed CT.

# $\perp\top^*$ -Mod(Snomed CT)	# \mathcal{M}_p^c (Projection Module)
1 / 116 / 2.0	1 / 4 / 1.0

Table 6.1 Sizes of $\perp\top^*$ -modules of Snomed CT and Projection Modules of Snomed CT regarding MeSH (minimal / maximal / median)

¹<http://bioportal.bioontology.org/ontologies/MESH>

²<http://www.ihtsdo.org/snomed-ct/>

Chapter 7

Conclusion and Future Work

7.1 Conclusion

In this thesis, we presented a framework to assist users to extract knowledge from terminologies that are formulated in the Description Logics \mathcal{ELH} and \mathcal{ELH}^r . We provided three different approaches catering for users' demands: extracting minimal basic subsumption modules (Chapter 4) that preserve subsumption queries over the terms of interest, computing best excerpts (Chapter 5) that best capture users' interest with a size restriction, and computing projection modules of a target ontology (Chapter 6) that entail subsumption, instance and conjunctive queries from a reference ontology. Computing best excerpts and minimal subsumption modules (glass-box approach) is based on a notion called subsumption justification. We presented algorithms for computing subsumption justifications of an acyclic \mathcal{ELH} - and \mathcal{ELH}^r -terminology w.r.t. a signature (Chapter 3). The black-box approach for computing minimal modules is language-independent. However, it relies on an external tool for computing the logical difference between ontologies. At present, the state-of-the-art tool for computing logical differences can only handle (cyclic) \mathcal{ELH}^r -terminologies. For computing projection modules, we proposed a new notion called projection justification. Contrary to subsumption justifications that only consider one ontology, a projection justification is related to two ontologies. A projection justification is a subset of a target ontology obtained from projecting into the target ontology all subsumption entailments of a reference ontology. We presented the algorithms for computing projection justifications for three different types of queries (subsumption, instance and conjunctive queries) and proved their correctness. A summary of the computational complexity of the algorithms developed in this thesis is presented in Table 7.1. In general, subsumption justifications, minimal modules, projection justifications and projection modules can all be computed in EXPTIME in the size of the input. However, computing best excerpts runs in NEXPTIME since the input clauses of the partial Max-SAT encoding can be exponentially large, which in turn is due to fact that the number of subsumption justifications is exponentially bounded. We implemented Java-prototypes of our algorithms and we evaluated them on large biomedical ontologies. The results show that the

approaches we propose in this thesis can assist users to extract knowledge that is relevant to their interests from description logic terminologies.

Conditions	Subsumption Justification	Minimal Module	Best Excerpt	Projection Justification	Projection Module
General Case	EXP	EXP	NEXP	EXP	EXP

Table 7.1 Complexity Results for \mathcal{ELH}^r -Terminologies

7.2 Future Work

We conclude the thesis by discussing possible directions for future work.

Subsumption Justification We would like to evaluate our algorithms for computing subsumption justifications for \mathcal{ELH}^r -terminologies after an efficient tool for computing justifications for \mathcal{ELH}^r -terminologies has been developed. Moreover, we expect that the algorithms for computing subsumption justifications could be extended to deal with cyclic terminologies and general \mathcal{EL} -TBoxes. A complete complexity analysis is also expected.

Basic Subsumption Modules In the near future, we plan to generate a benchmark for basic subsumption modules on \mathcal{ELH} - and \mathcal{ELH}^r -fragment of Snomed CT and NCI ontology. We also expect to be able to compute basic subsumption modules of ontologies formulated in more expressive description logics. One option is to extend the algorithm to compute subsumption justifications as mentioned above. We can also benefit from the research on logical difference and uniform interpolation, as the black-box algorithm is language-independent. It has been shown that our black-box algorithm can be used to compute subsumption modules for \mathcal{ALCH} -ontologies in [32].

Best Ontology Excerpts Currently, the notion of best excerpts that has been investigated is based on the metric of logical difference. We would like to explore alternative metrics to enrich the family of ontology excerpts. One possibility is based on restricted reasoning tasks, such as instance retrieval of named concepts [41], named concept subsumption [39], or unsatisfiable concepts [18]. The intuition is that these simplified reasoning tasks are often the main concern in real applications. The current definition of ontology excerpt insists on keeping the original form of the axioms. Another interesting direction to generalise the notion of ontology excerpts is to allow for modified axioms. For this, a procedure similar to *knowledge compilation* appears to be promising.

Computing optimal ontology excerpts is often computationally too expensive. An alternative is to compute approximate ontology excerpts. For this, it is possible to benefit from approximating algorithms for Constraint-Satisfaction-Problems once the previous task can be conducted. A different sort of technique for extracting quasi-optimal excerpts has been proposed by using a model from *Information Retrieval* [9]. The empirical study shows that it gives non-trivial results compared to

a direct enumeration or random selection on large real-world ontologies but nevertheless fails to find the optimal excerpts. To improve this technique, a possibility is to take into account the logical contribution of each single axiom into the existing approximating model, which could boost its performance.

Projection Modules We expect that the algorithms can be extended to deal with cyclic terminologies and possibly general \mathcal{ELH}^r -TBoxes and consider algorithms for ranking different projection modules, via weighted signatures for instance. Moreover, we would like to find applications for projection modules. For example, developing a uniform upper ontology to guide ontology engineers to build ontologies that satisfy certain requirements.

Bibliography

- [1] Alsubait, T., Parsia, B., and Sattler, U. (2014). Measuring similarity in ontologies: A new family of measures. In *Proceedings of EKAW'14: the 19th International Conference on Knowledge Engineering and Knowledge Management*, pages 13–25.
- [2] Arif, M. F., Mencía, C., Ignatiev, A., Manthey, N., Peñaloza, R., and Marques-Silva, J. (2016). BEACON: An efficient SAT-based tool for debugging EL^+ -ontologies. In *Proceedings of SAT'16: the 19th International Conference on the Theory and Applications of Satisfiability Testing*, pages 521–530.
- [3] Baader, F., Calvanese, D., McGuinness, D. L., Nardi, D., and Patel-Schneider, P. F., editors (2010). *The description logic handbook: theory, implementation, and applications*. Cambridge University Press, 2 edition.
- [4] Baader, F., Peñaloza, R., and Suntisrivaraporn, B. (2007a). Pinpointing in the description logic EL. In *Proceedings of KI'07: the 30th Annual German Conference on Artificial Intelligence*, volume 4667 of *LNAI*, pages 52–67.
- [5] Baader, F., Peñaloza, R., and Suntisrivaraporn, B. (2007b). Pinpointing in the description logic \mathcal{EL} . In *Proceedings of DL'07: the 20th International Workshop on Description Logics*.
- [6] Botoeva, E., Konev, B., Lutz, C., Ryzhikov, V., Wolter, F., and Zakharyashev, M. (2017). Inseparability and conservative extensions of description logic ontologies: A survey. In *Reasoning Web Summer School 2016*, volume 9885 of *LNCS*, pages 27–89.
- [7] Botoeva, E., Kontchakov, R., Ryzhikov, V., Wolter, F., and Zakharyashev, M. (2016a). Games for query inseparability of description logic knowledge bases. *Artificial Intelligence*, 234:78 – 119.
- [8] Botoeva, E., Lutz, C., Ryzhikov, V., Wolter, F., and Zakharyashev, M. (2016b). Query-based entailment and inseparability for alc ontologies. In *Proceedings of IJCAI'16: the 25th International Joint Conference on Artificial Intelligence*, pages 1001–1007. AAAI Press.
- [9] Chen, J., Ludwig, M., Ma, Y., and Walther, D. (2015). Towards extracting ontology excerpts. In *Proceedings of KSEM'15: the 8th International Conference on Knowledge Science, Engineering and Management*, pages 78–89.
- [10] Chen, J., Ludwig, M., and Walther, D. (2016). On computing minimal el-subsumption modules. In *Proceedings of the Joint Ontology Workshops 2016 Episode 2: The French Summer of Ontology co-located with the 9th International Conference on Formal Ontology in Information Systems (FOIS 2016)*.
- [11] Del Vescovo, C. and Peñaloza, R. (2014). Dealing with ontologies using cods. In *Proceedings of DL'14: the 27th International Workshop on Description Logics*, volume 1193 of *CEUR Workshop Proceedings*, pages 157–168.

- [12] Ecke, A., Ludwig, M., and Walther, D. (2013). The concept difference for \mathcal{EL} -terminologies using hypergraphs. In *Proceedings of DChanges'13: the 1st International Workshop on: Document Changes: Modeling, Detection, Storage and Visualization*.
- [13] Euzenat, J. and Shvaiko, P. (2013). *Ontology Matching, Second Edition*.
- [14] Fu, Z. (2007). *Extending the Power of Boolean Satisfiability: Techniques and Applications*. PhD thesis, Princeton University.
- [15] Gatens, W., Konev, B., and Wolter, F. (2014). Lower and upper approximations for depleting modules of description logic ontologies. In *Proceedings of ECAI'14: the 21st European Conference on Artificial Intelligence*, pages 345–350.
- [16] Grau, B. C., Horrocks, I., Kazakov, Y., and Sattler, U. (2008). Modular reuse of ontologies: Theory and practice. *Journal of Artificial Intelligence Research*, 31(1):273–318.
- [17] Hofmann, M. (2005). Proof-theoretic approach to description-logic. In *Proceedings of LICS'05: the 20th Annual IEEE Symposium on Logic in Computer Science*, pages 229–237.
- [18] Jiménez-Ruiz, E. and Grau, B. C. (2011). Logmap: Logic-based and scalable ontology matching. In *Proceeding of ISWC'11: the 10th International Semantic Web Conference*, pages 273–288.
- [19] Jung, J. C., Lutz, C., Martel, M., and Schneider, T. (2017). Query conservative extensions in horn description logics with inverse roles. In *Proceedings of IJCAI'17: the 26th International Joint Conference on Artificial Intelligence*, pages 1116–1122.
- [20] Kalyanpur, A., Parsia, B., Horridge, M., and Sirin, E. (2007). Finding all justifications of OWL DL entailments. In *Proceedings of ISWC'07 & ASWC'07: the 6th International Semantic Web Conference and 2nd Asian Semantic Web Conference*, volume 4825 of LNCS, pages 267–280.
- [21] Kalyanpur, A., Parsia, B., Sirin, E., and Hendler, J. A. (2005). Debugging unsatisfiable classes in OWL ontologies. *Journal of Web Semantic*, 3(4):268–293.
- [22] Kazakov, Y. and Skocovský, P. (2018). Enumerating justifications using resolution. In *Proceedings of IJCAR'18: the 9th International Joint Conference on Automated Reasoning*, pages 609–626.
- [23] Konev, B., Ludwig, M., Walther, D., and Wolter, F. (2012a). The logical difference for the lightweight description logic EL. *Journal of Artificial Intelligence Research*, 44:633–708.
- [24] Konev, B., Ludwig, M., and Wolter, F. (2012b). Logical difference computation with CEX2.5. In *Proceedings of IJCAR'12: the 6th International Joint Conference on Automated Reasoning*, pages 371–377.
- [25] Konev, B., Lutz, C., Walther, D., and Wolter, F. (2008a). Semantic modularity and module extraction in description logics. In *Proceedings of ECAI'08: the 18th European Conference on Artificial Intelligence*, pages 55–59.
- [26] Konev, B., Lutz, C., Walther, D., and Wolter, F. (2009). *Formal Properties of Modularisation*, pages 25–66. Springer Berlin Heidelberg, Berlin, Heidelberg.
- [27] Konev, B., Lutz, C., Walther, D., and Wolter, F. (2013). Model-theoretic inseparability and modularity of description logic ontologies. *Artificial Intelligence*, 203:66–103.

- [28] Konev, B., Walther, D., and Wolter, F. (2008b). The logical difference problem for description logic terminologies. In *Proceedings of IJCAR'08: the 4th International Joint Conference on Automated Reasoning*, pages 259–274.
- [29] Kontchakov, R., Pulina, L., Sattler, U., Schneider, T., Selmer, P., Wolter, F., and Zakharyashev, M. (2009). Minimal module extraction from DL-Lite ontologies using QBF solvers. In *Proceedings of DL'09: the 21st International Workshop on Description Logics*, pages 836–841.
- [30] Kontchakov, R., Wolter, F., and Zakharyashev, M. (2008). Can you tell the difference between dl-lite ontologies? In *Proceedings of KR'08: the 11th International Conference on Principles of Knowledge Representation and Reasoning*, pages 285–295.
- [31] Kontchakov, R., Wolter, F., and Zakharyashev, M. (2010). Logic-based ontology comparison and module extraction, with an application to DL-Lite. *Artificial Intelligence*, 174(15):1093–1141.
- [32] Koopmann, P. and Chen, J. (2017). Computing ALCH-subsumption modules using uniform interpolation. In *Proceedings of SOQE'17: the Workshop on Second-Order Quantifier Elimination and Related Topics*, pages 51–66.
- [33] Lehmann, K. and Turhan, A. (2012). A framework for semantic-based similarity measures for ELH-concepts. In *Proceedings of JELIA'12: the 13th European Conference on Logics in Artificial Intelligence*, pages 307–319.
- [34] Ludwig, M. and Walther, D. (2014). The logical difference for ELHr-terminologies using hypergraphs. In *Proceedings of ECAI'14: the 21st European Conference on Artificial Intelligence*, pages 555–560.
- [35] Lutz, C. and Wolter, F. (2010). Deciding inseparability and conservative extensions in the description logic EL. *Journal of Symbolic Computation*, 45(2):194–228.
- [36] Martín-Recuerda, F. and Walther, D. (2014). Fast modularisation and atomic decomposition of ontologies using axiom dependency hypergraphs. In *Proceedings of ISWC'14: the 13th International Semantic Web Conference*, pages 49–64.
- [37] Presutti, V., Blomqvist, E., Daga, E., and Gangemi, A. (2012). Pattern-based ontology design. In *Ontology Engineering in a Networked World*, pages 35–64.
- [38] Reiter, R. (1987). A theory of diagnosis from first principles. *Artificial Intelligence*, 32(1):57–95.
- [39] Ren, Y., Pan, J. Z., and Zhao, Y. (2010). Soundness preserving approximation for tbox reasoning. In *Proceedings of AAAI'10: the 24th AAAI Conference on Artificial Intelligence*.
- [40] Romero, A. A., Kaminski, M., Grau, B. C., and Horrocks, I. (2016). Module extraction in expressive ontology languages via datalog reasoning. *Journal of Artificial Intelligence Research*, 55:499–564.
- [41] Rudolph, S., Tserendorj, T., and Hitzler, P. (2008). What is approximate reasoning? In *Proceedings of RR'08: the 2nd International Conference on Web Reasoning and Rule Systems*, pages 150–164.
- [42] Sattler, U., Schneider, T., and Zakharyashev, M. (2009). Which kind of module should I extract? In *Proceedings of DL'09: the 21st International Workshop on Description Logics*, volume 477.

- [43] Schlicht, A. and Stuckenschmidt, H. (2007). Criteria-based partitioning of large ontologies. In *Proceedings of K-CAP'07: the 4th International Conference on Knowledge Capture*, pages 171–172.
- [44] Sinz, C. (2005). Towards an optimal cnf encoding of boolean cardinality constraints. In *Proceedings of CP'05: the 11th International Conference on Principles and Practice of Constraint Programming*, pages 827–831.
- [45] Suárez-Figueroa, M. C. and Gómez-Pérez, A. (2012). Ontology requirements specification. In *Ontology Engineering in a Networked World*, pages 93–106.
- [46] Suntisrivaraporn, B., Qi, G., Ji, Q., and Haase, P. (2008). A modularization-based approach to finding all justifications for OWL DL entailments. In *Proceeding of ASWC'08: the 3rd Asian Semantic Web Conference*, pages 1–15.
- [47] Troullinou, G., Kondylakis, H., Daskalaki, E., and Plexousakis, D. (2015). RDF digest: Efficient summarization of RDF/S KBs. In *Proceedings of ESWC'15: the 12th European Semantic Web Conference*, volume 9088 of *Lecture Notes in Computer Science*, pages 119–134.
- [48] Vescovo, C. D., Gessler, D., Klinov, P., Parsia, B., Sattler, U., Schneider, T., and Winget, A. (2011a). Decomposition and modular structure of biportal ontologies. In *Proceedings of ISWC'11: the 10th International Semantic Web Conference*, volume 7031 of *Lecture Notes in Computer Science*, pages 130–145.
- [49] Vescovo, C. D., Parsia, B., and Sattler, U. (2012). Logical relevance in ontologies. In *Proceedings of DL'12: the 25th International Workshop on Description Logics*, volume 846.
- [50] Vescovo, C. D., Parsia, B., Sattler, U., and Schneider, T. (2011b). The modular structure of an ontology: Atomic decomposition. In *Proceedings of IJCAI'11 : the 22nd International Joint Conference on Artificial Intelligence*, pages 2232–2237.
- [51] Zhang, X., Cheng, G., and Qu, Y. (2007). Ontology summarization based on rdf sentence graph. In *Proceedings of WWW'07: the 16th International Conference on World Wide Web*, pages 707–716.
- [52] Zhou, Z., Qi, G., and Suntisrivaraporn, B. (2013). A new method of finding all justifications in OWL 2 EL. In *Proceedings of WI'13: the IEEE/WIC/ACM International Conferences on Web Intelligence*, pages 213–220.

Titre : Extraction de connaissances à partir de terminologies en logique de description.

Mots clés : Représentation et raisonnement des connaissances, logique de description, Web sémantique, modularité d'ontologies, différence logique.

Résumé : Un nombre croissant d'ontologies de grande taille sont en cours de développement et rendues disponibles, par exemple dans des référentiels tels que le bioportail de NCBO. Garantir l'accès aux connaissances les plus pertinentes pour les utilisateurs contenues dans des ontologies est un défi important. Dans ce travail, nous abordons ce défi en proposant trois approches différentes pour extraire des connaissances des ontologies en logique de description: extraire les modules minimaux d'ontologies (c'est-à-dire les sous-ontologies minimales pour l'inclusion ensembliste qui préservent toutes les implications sur un vocabulaire donné) ; calculer les meilleurs extraits d'ontologie (un certain nombre, petit, d'axiomes qui capturent au mieux la connaissance du vocabulaire tout en permettant un certain degré de perte sémantique) ; et déterminer les modules de projection (sous-ontologies d'une ontologie cible qui impliquent des requêtes de subsomption, d'instance ou conjonctives qui découlent d'une ontologie de référence). Pour chacune de ces approches, nous souhaitons extraire non seulement une, mais toutes les instances de la notion correspondante de module. Pour le calcul des modules minimaux et des meilleurs extraits, nous introduisons la notion de justification de la subsomption en tant que généralisation de la notion de justification (un ensemble minimal d'axiomes nécessaires pour préserver une conséquence logique donnée) afin de capturer la connaissance déduite par subsomption sur le vocabulaire. De même, pour le calcul des modules de projection, nous introduisons la notion de justifications de projection qui préservent les réponses à l'un des trois types de requête impliquées par une ontologie de référence. Enfin, nous évaluons nos approches à l'aide d'un prototype d'implémentation des algorithmes sur de grandes ontologies.

Title : Knowledge Extraction from Description Logic Terminologies

Keywords : Knowledge Representation and Reasoning, Description Logics, Semantic Web, Ontology Modularity, Logical Difference.

Abstract : An increasing number of large ontologies are being developed and made available, e.g., in repositories such as the NCBO Bioportal. Ensuring access to the knowledge contained in ontologies that is most relevant to users has been identified as an important challenge. In this work, we tackle this challenge by proposing three different approaches to extracting knowledge from Description Logic ontologies: extracting minimal ontology modules (i.e., sub-ontologies that are minimal w.r.t. set inclusion while still preserving all entailments over a given vocabulary); computing best ontology excerpts (a certain, small number of axioms that best capture the knowledge about the vocabulary while allowing for a degree of semantic loss); and determining projection modules (sub-ontologies of a target ontology that entails subsumption, instance or conjunctive queries that follow from a reference ontology). For each of these approaches, we are interested in extracting not only one but all instances of the module notion. For computing minimal modules and best excerpts, we introduce the notion of subsumption justification as a generalisation of the notion of a justification (a minimal set of axioms needed to preserve a given logical consequence) to capture the subsumption knowledge over the vocabulary. Similarly, for computing projection modules, we introduce the notion of projection justifications that preserve the answers to one of three query types as given by a reference ontology. Finally, we evaluate our approaches using a prototype implementation of the algorithms on large ontologies.

