Tractable Approximate Deduction for OWL ☆

Jeff Z. Pan[1], Yuan Ren, Yuting Zhao

Department of Computing Science, University of Aberdeen
AB24 3UE, Aberdeen, UK

Abstract

Today's ontology applications require efficient and reliable description logic (DL) reasoning services. Expressive DLs usually have high worst case complexity while tractable DLs are restricted in terms of expressive power. This brings a new challenge: can users use expressive DLs to build their ontologies and still enjoy the efficient services as in tractable languages? Approximation has been considered as a solution to this challenge; however, traditional approximation approaches have limitations in terms of performance and usability. In this paper, we present a tractable approximate reasoning framework for OWL 2 that improves efficiency and guarantees soundness. Evaluation on ontologies from benchmarks and real-world use cases shows that our approach can do reasoning on complex ontologies efficiently with a high recall.

Keywords:
Ontology, Approximation, OWL 2, Reasoning

1. Introduction

With the growing volume and complexity of ontologies [27] and large-scale linked data [2] available, there is a pressing need for efficient ontology reasoning services. Modern ontology language OWL 2, the second version of OWL (Web Ontology Language), is based on a family of different Description Logics (DLs) [3]. For example, OWL 2 DL, the most expressive and decidable OWL, is based on $\mathcal{SROIQ}$ [30], a very expressive but complex DL. Its three tractable profiles OWL 2 EL, OWL 2 QL and OWL 2 RL are based on simpler but less expressive formalisms $\mathcal{EL}^{++}$ [1], DL-Lite$_R$ [11] and DLP [26], respectively. Such a spectrum of DLs leads to different approaches of ontology reasoning.

One approach is to develop fully-fledged universal algorithms that can be applied on any decidable DL. For example, tableau-based algorithms [17, 31, 32] with complexity up to N2EXPTIME-complete [40] can provide TBox (terminology) reasoning services, such as classification for very expressive DLs including $\mathcal{SROIQ}$. ABox (assertion) reasoning, such as checking whether a

[1] Corresponding author. Email address: jeff.z.pan@abdn.ac.uk. Phone number: +44 1224 274449.

☆This paper is an extended version of our previous work. Particularly, the technical work presented in Sec. 5 is based on our AAAI2010 paper [65] but revised and extended with support to more expressive power. The technical work presented in Sec. 6 is based on our DL workshop 2010 paper [66] but significantly improved with different completion rule sets and completeness proofs. We refer to the corresponding summary sections for discussion of gaps.

[2] `http://wifo5-03.informatik.uni-mannheim.de/lodcloud/state/`

given individual is an instance of a given concept, is usually realised by extensions of TBox algorithms [33], or by reduction to TBox reasoning [69]. To reduce non-determinism in tableau algorithms, different absorption techniques [80, 79, 36] have been extensively investigated, which are later generalised by the hypertableau calculus [59]. Decomposition-based approaches have been developed to improve performance of ontology classification [67] and conjunctive query answering [18]. Other work [37] reduces ontologies into disjunctive datalog to provide dedicated ABox reasoning. Due to the high worst case complexity of the algorithms employed, some of the expressive and difficult ones, such as the Foundational Model of Anatomy Constitutional ontology (FMA Constitutional), can hardly be classified by any fully-fledged OWL reasoner in reasonable time. Given the current efforts of ontology construction, it might not take long before many other even larger and more complicated ontologies appear and go beyond the capability of existing DL reasoners.

Another approach to ontology reasoning is to develop dedicated and very efficient algorithms for specific light-weight DLs/profiles. For example, reasoning in $\mathcal{EL}$ family is PTIME-complete [1] and can be executed even in parallel [42]. Similar consequence-based algorithms can further be extended to support non-tractable DLs such as Horn-$\mathcal{SHIQ}$ [41] and $\mathcal{ALCH}$ [73]. However, such light-weight DLs usually impose strong expressiveness restrictions that are difficult to be satisfied in real-world scenario. First of all, many applications inherently require highly expressive ontology languages. For example, in software engineering extending UML (Unified Modeling Language) or DSL (Domain Specific Language) with OWL [63, 84] requires combinations of many OWL constructors not supported by any single profile. Secondly, forcing users to use profiles in ontology engineering will under-specify the ontologies because they have to give up the knowledge that can only be represented with more expressive languages. Last but not least, in a linked data scenario, it is almost impossible to guarantee that all the connected ontologies belong to the same profile. As a consequence, many real-world ontologies are beyond any tractable profile and require an OWL 2 DL reasoner to reason with.

Like any other software, ontology engineering involves development and deployment. In the development stage, ontology engineers would like to construct their ontologies with as much expressiveness as possible. While in the deployment stage, ontology users want to reason with the ontologies as efficient as possible. Given the trade-off between expressiveness and performance, it is difficult for either of the above approaches to satisfy both requirements. This brings a new challenge: can users use OWL 2 DL to build their ontologies and still enjoy the efficient reasoning as in tractable profiles?

To answer this challenge, *approximation*-based approaches have been studied and evaluated. Essentially, approximation approaches bring a new dimension–quality, in terms of completeness and soundness of reasoning, into the trade-off between expressiveness and performance, attempting to strike a balance among the three. This is motivated by the fact that real-world knowledge and data are hardly perfect or complete. Hence sacrificing some theoretical quality to achieve more engineering flexibility and practical efficiency can be desirable in many scenarios. Some approximation approaches apply the idea of *knowledge compilation* [72] and *language weakening* by transforming ontologies in complex DLs into other easier-to-reason-with formalisms [61, 16, 51, 6]. Some apply the idea of *approximate deduction* by treating complex expressions or constructors in ontologies or queries as simpler ones [70, 68, 25, 76, 83, 28, 81]. There is also approach that improves reasoning efficiency by identifying and removing difficult sub-ontologies, known as hotspots [24]. However, most of the existing approximation approaches still have important limitations in the sense that they cannot always guarantee a reduction of complexity, or they rely on other reasoners that impose restrictions on expressive power, or they require non-trivial off-line computation.

One important aspect on approximation is quality of approximation. Pan and Thomas [61] proposed a notion of strongest weaker approximations (or semantic approximations) of DL ontologies, which come with a conditional completeness result: their approximation guarantees both soundness and completeness for database style conjunctive quires (i.e., queries in which variables could only be bounded to named individuals). In other words, database style conjunctive queries can not tell the difference between an original ontology and its semantic approximation. However, computing such optimal approximation is expense. Reasoning is needed to compute semantic approximations; therefore, the construction of semantic approximations is usually done off line. In fact, for ontologies in intractable DLs, it is in general not possible to compute in polynomial time, the strongest weaker approximations in tractable DLs. Otherwise, the classification of the original ontologies can be achieved in polynomial time by classifying the tractable approximations.

In this paper we present *recursive syntactic approximation*, a less expensive yet soundness preserving approximate reasoning approach to supporting TBox and ABox reasoning in OWL 2 DL. The hypothesis behind syntactic approximation is that many real-world ontologies, although use expressive ontology constructors, mostly use them in a way that their interactions can be captured by tractable algorithms. And consequently, it is possible to develop very efficient algorithms to reason with these expressive ontologies without losing too much completeness. Particularly, our approach reduces reasoning complexity from N2EXPTIME-hard to PTIME-complete and the reasoning result is reasonably complete (mostly from 97% to 100%) according to our evaluations. More precisely, our contributions are:

1. We losslessly transform an OWL 2 DL ontology into an $\mathcal{EL}^{++}$ ontology with additional data structures to maintain non-$\mathcal{EL}^{++}$ semantics, such as complement, cardinality and inverse role, then extend the $\mathcal{EL}^{++}$ reasoning with approximate deduction rules to perform materialisation for both TBox and ABox. These additional rules are designed to partially recover the non-$\mathcal{EL}^{++}$ semantics. We show that such materialisation services are all tractable (cf. Theorem 4 and Theorem 10) and soundness-preserving (cf. Theorem 5 and Theorem 11).
2. To improve the performance on large data sets, we present and discuss several more and more fine-grained ABox reasoning optimisations, all of which are tractable and soundness-preserving. We also show that these optimisations will not affect the completeness of approximation in materialisation (cf. Theorem 12, Theorem 14 and Theorem 15).
3. To evaluate the performance of our approach, we implement the presented approximate reasoning mechanisms in the *REL* reasoner (one of the component reasoners in TrOWL [78, 62, 60]) and conduct comprehensive evaluations on both benchmark ontologies and real-world ontologies. Results show that REL outperforms fully-fledged OWL 2 DL reasoners and yields high recall on all tested ontologies. This further verifies our hypothesis. The ontologies we evaluated in this paper and the REL reasoner we implemented are available for reproduction of our results. They can be downloaded from `http://homepages.abdn.ac.uk/jeff.z.pan/pages/link/AIJReproductionPackage.zip`.

Compared to our previous work [65, 66] on which this paper is based, the current paper revises and extends the definition of transformations (cf. Def. 4 and Def. 8) and the rule sets extensively (from Table 5 to Table 10). Preservation of reasoning results by transformations is discussed (cf. Theorem 2 and Theorem 8). We also present formal proofs on the equivalence of different ABox optimisation approaches in materialisation. The evaluation has also be extended to support the new technical extensions. More extensive discussion on the relation to existing works, incompleteness and partial completeness (Theorem 17) is included.

The rest of this paper is organised as follows: in Sec. 2 we introduce preliminaries about DLs. In Sec. 3 we review related work and discuss the technical challenges of existing reasoning approaches. In Sec. 4 we outline our approach with examples. In Sec. 5 and 6 we present our approaches to syntactic approximation. In Sec. 7 we present evaluation results of our approaches. In Sec. 8 we discuss the relation to existing works, incompleteness and partial completeness of our approach. In Sec. 9 we conclude the paper. As our approach is soundness preserving, it is important to clarify the the charaterisation of incompleteness of our approach. For this purpose, we have included, in Sec. 6.4, an example (Example 8) to illustrate motivation of our approach in the light of the trade-off between incompleteness and intractability. In Sec. 7, we have detailed discussions about recall, including the causes of incompleteness found in the evaluation. In Sec. 8.2, we show that fixing certain types of incompleteness issues will lead to intractability issues. In Sec. 8.3, we show how one could exploit above trade-off between incompleteness and intractability for a conditional completeness approach for the ELHI DL.

For the sake of conciseness, we only present sketchy proofs for formal results throughout the paper. Complete proofs are deferred to the Appendix after the end of the paper.

2. Background Knowledge

In this section, we briefly introduce description logics and ontology. To facilitate the presentation of syntactic approximation and its formal properties, we focus on two DL dialects, namely $\mathcal{SROIQ}$ and $\mathcal{EL}^{++}$ and present their syntax, semantics and inference problems. For the sake of conciseness, in this paper we will omit the DL elements that are not relevant to our technical work (e.g. datatypes).

The description logics is a family of logic formalisms that describe the domain of discourse with so called *concept* and *role expressions* and their *instances*. A *signature* is a triple $\Sigma = (\mathcal{CN}, \mathcal{RN}, \mathcal{IN})$ consisting of three mutually disjoint sets of named concepts $\mathcal{CN}$, named roles $\mathcal{RN}$and individuals $\mathcal{IN}$. In DLs, *concept expressions* (or *concepts* for short) and *roles expressions* (or *roles* for short) can be constructed inductively from a signature with different constructors. The kinds of constructors that can be used and the ways they can be combined determine the level of expressiveness of the particular DL. Different DLs have different expressive power.

2.1. $\mathcal{SROIQ}$ Syntax

In $\mathcal{SROIQ}$, the underpinning logic of OWL 2 DL [57], the set of roles is $\mathcal{RN} \cup \{R^-|R \in \mathcal{RN}\}$. On this set, the inverse role function $Inv(.)$ is defined as $Inv(R) = R^-$ and $Inv(R^-) = R$ where $R \in \mathcal{RN}$. The set of role axioms, namely the RBox $\mathcal{R}$, is a finite set of either **Inclusion** axioms of form $R_1 \sqsubseteq R_2$ or $R_1 \circ R_2 \sqsubseteq R_3$, or **Disjointness** axioms of form $Dis(S_1, S_2)$, where R, $R_1 \ldots R_3$ are roles, and S_1, S_2 are *simple roles* which we will define later.

In the full specification of $\mathcal{SROIQ}$ [30], there are other role axioms such as *reflexivity* and *irreflexivity*, which we will omit in this paper and refer our readers to that paper for more detail. Role inclusions axioms (**RIA**s) with role chains of more than two roles, such as $R_1 \circ R_2 \circ \cdots \circ R_n \sqsubseteq R_{n+1}$, can also be normalised into binary role chain role inclusions as we defined above in linear time [4]. Two roles R_1 and R_2 are **Equivalent** if they mutually include each other, denoted by $R_1 \equiv R_2$. Besides, **Symmetry** $Sym(R)$ can be defined as $R \equiv Inv(R)$.

Asymmetry $Asy(S)$ can be defined as $Dis(R, Inv(R))$. **Transitivity** $Trans(R)$ can be defined as $R \circ R \sqsubseteq R$.

The notion of **Simple Role** is inductively defined in $\mathcal{R}$ as follows:

- $S \in \mathcal{RN}$ is simple if S does not appear on the RHS (right hand side) of any RIA in $\mathcal{R}$;
- $S \in \mathcal{RN}$ is simple if $Inv(S)$ is simple;
- $S \in \mathcal{RN}$ is simple if for each $R \sqsubseteq S \in \mathcal{R}$, R is simple;

Given $\Sigma = (\mathcal{CN}, \mathcal{RN}, \mathcal{IN})$, the set of $\mathcal{SROIQ}$ concepts is the smallest set containing every concept of the following form: *atomic concept* A, *nominal* $\{a\}$, the *top concept* $\top$, the *bottom concept* $\bot$, *complement* $\neg C$, *conjunction* $C \sqcap D$, *existential restriction* $\exists R.C$ and *at-least restriction* $\geq nS.C$, where $A \in \mathcal{CN}$, $a \in \mathcal{IN}$, C and D are concepts, R is a role, S is a simple role, and n is a non-negative integer.

Conventionally, disjunction $C \sqcup D$, universal restriction $\forall R.C$ and at-most restriction $\leq nR.C$ are used to abbreviate $\neg(\neg C \sqcap \neg D)$, $\neg\exists R.\neg C$ and $\neg \geq (n+1)R.C$, respectively. Enumeration $\{a_1, a_2, \ldots, a_n\}$ can be regarded as abbreviation of $\{a_1\} \sqcup \{a_2\} \sqcup \cdots \sqcup \{a_n\}$.

A TBox $\mathcal{T}$ is a finite set of general concept inclusions (**GCI**s) of the form $C \sqsubseteq D$. Two concepts C and D are **Equivalent** if they mutually include each other, denoted by $C \equiv D$. An ABox $\mathcal{A}$ is a finite set of concept assertion $a : C$, role assertion $(a, b) : R$, negative role assertion $(a, b) : \neg R$, individual equality $a \doteq b$ and individual inequality $a \not\doteq b$, for C a concept, R a role and $a, b \in \mathcal{IN}$. A $\mathcal{SROIQ}$ knowledge base (or ontology) $\mathbf{K}$ is a triple $(\mathcal{T}, \mathcal{R}, \mathcal{A})$. In what follows, for sake of simplicity, we include the RBox $\mathcal{R}$ into TBox $\mathcal{T}$, and simplify a $\mathcal{SROIQ}$ knowledge base as tuple $(\mathcal{T}, \mathcal{A})$ in which $\mathcal{T}$ contains all the GCIs and role axioms.

2.2. $\mathcal{EL}^{++}$ Syntax

The $\mathcal{EL}$ family is dedicated for large TBox reasoning and has been widely applied in some largest ontologies, e.g. SNOMED CT [75]. $\mathcal{EL}^{++}$, the underpinning logic of OWL 2 EL [56], supports the following concept constructors:

$$\top \mid \bot \mid A \mid C \sqcap D \mid \exists r.C \mid \{a\},$$

in which $A \in \mathcal{CN}$, C and D are $\mathcal{EL}^{++}$ concepts and $a \in \mathcal{IN}$. Different from $\mathcal{SROIQ}$, role r can only be an atomic role. Similar as $\mathcal{SROIQ}$, an $\mathcal{EL}^{++}$ ontology $\mathbf{K}$ is a tuple $(\mathcal{T}, \mathcal{A})$, in which $\mathcal{T}$ is a TBox consisting of finite GCIs and RIAs, and $\mathcal{A}$ is an ABox consisting of finite concept assertions, role assertions, individual equality and inequality axioms, but without negative role assertions. $\mathcal{EL}^{++}$ was further extended by Franz Baader et. al. [2] to support range restrictions of form $\top \sqsubseteq \forall r.C$, where C is a valid $\mathcal{EL}^{++}$ concept. Some other features included in the OWL 2 EL profile, such as **Reflexivity**, will be omitted in this paper.

Krötzsch et al. [44] showed that using nominals liberally in $\mathcal{EL}^{++}$ ontologies will make reasoning more difficult. To address this problem, the authors later identified the nominal-safe criteria in which nominals can be used without increasing the difficulty of reasoning [43]. An $\mathcal{EL}^{++}$ ontology is nominal-safe if all GCIs are of form $C_{NS} \sqsubseteq C_S$, where

$$C_{NS} = \{a\} \mid C_S \text{ and } C_S = \top \mid \bot \mid A \mid C_S \sqcap C_S \mid \exists r.C_S \mid \exists r.\{a\}.$$

Note that although axiom $\{a\} \equiv \{b\}$ and $\{a\} \sqcap \{b\} \sqsubseteq \bot$ are not included in the above nominal-safety syntax, they can be equivalently rewritten as $a \doteq b$ and $a \not\doteq b$, respectively, so we regard these kinds of axioms as nominal-safe. In this paper, we restrict ourselves to the kind of $\mathcal{SROIQ}$ ontologies whose approximate reasoning closures, as we will define in later sections, are always nominal-safe $\mathcal{EL}^{++}$ ontologies (Def. 6, 7 and Theorem. 6). This restriction is due to the following reasons:

1. As indicated by Markus et al., reasoning procedure of nominal-safe $\mathcal{EL}^{++}$ ontologies is simpler and more efficient.
2. Many real-world ontologies, including the majority of ontologies we used for evaluation in this paper, satisfy such a nominal-safe $\mathcal{SROIQ}$ criteria. More detailed analysis will be presented in Sec. 7.3.

Without loss of generality, in what follows, we assume all the concepts to be in their negation normal forms (NNF)[3] and use $\sim C$ to denote the NNF of $\neg C$. We also slightly abuse the notion of atomic concept to include $\top$, $\bot$ and all nominals $\{a\}$ as well, i.e. $\mathcal{CN} := \mathcal{CN} \cup \{\top, \bot\} \cup \{\{a\} | a \in \mathcal{IN}\}$. Given a knowledge base $\mathbf{K}$, we also use $\mathcal{CN}_{\mathbf{K}}$ ($\mathcal{RN}_{\mathbf{K}}$, $\mathcal{IN}_{\mathbf{K}}$) to denote the set of atomic concepts (atomic roles, individuals) in $\mathbf{K}$ and use $\Sigma_{\mathbf{K}} = (\mathcal{CN}_{\mathbf{K}}, \mathcal{RN}_{\mathbf{K}}, \mathcal{IN}_{\mathbf{K}})$ to denote the signature of $\mathbf{K}$. Given an axiom α, we also use Σ_α to denote the signature of $\{\alpha\}$.

2.3. DL Semantics

The semantics of a DL is defined in terms of interpretations. An *interpretation* $\mathcal{I}$ is a pair $(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ where $\Delta^{\mathcal{I}}$ is a non-empty set and $\cdot^{\mathcal{I}}$ is a function that maps each atomic concept $A \in \mathcal{CN}$ to a subset $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, each atomic role $r \in \mathcal{RN}$ to a binary relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ and each individual a to an object $a^{I} \in \Delta^{\mathcal{I}}$. Interpretation function $\cdot^{\mathcal{I}}$ can be extended to complex concept and role expressions. Particularly, given an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, concepts C, D, roles R, S, and non-negative integer n, the interpretation of complex concept or role expressions used in the $\mathcal{SROIQ}$ and $\mathcal{EL}^{++}$ is inductively defined as in the upper part of Table 1, where $\sharp X$ denotes the cardinality of a set X. In addition to the original $\mathcal{SROIQ}$ role expressions, we also include the complement of a role R, denoted by $\neg R$. It is interpreted by the set of pairs of objects that are not included in $R^{\mathcal{I}}$. Such a expression will be used in approximation of role disjointness.

$\mathcal{I}$ is a model of $\mathbf{K}$, written $\mathcal{I} \models \mathbf{K}$, if it satisfies all axioms of $\mathbf{K}$ as shown in the lower part of Table 1. A knowledge base is *consistent* if it has a model. An axiom α is entailed by a knowledge base $\mathbf{K}$, written $\mathbf{K} \models \alpha$, iff all interpretations $\mathcal{I}$ of $\mathbf{K}$ satisfies α. A concept C is *satisfiable* w.r.t. a knowledge base $\mathbf{K}$ if there exists $\mathcal{I} \models \mathbf{K}$ and $C^{\mathcal{I}} \neq \emptyset$. In DLs there are several typical inference problems. In this paper, we are particularly interested in computing *materialisations* defined as follows:

Definition 1 ***(Materialisation)*** *For an ontology $\mathcal{O}$, its ontology materialisation is the set* $\{A \sqsubseteq B | A, B \in \mathcal{CN}_{\mathcal{O}}, \mathcal{O} \models A \sqsubseteq B\} \cup \{a : A | A \in \mathcal{CN}_{\mathcal{O}}, a \in \mathcal{IN}_{\mathcal{O}}, \mathcal{O} \models a : A\} \cup \{(a, b) : r | r \in \mathcal{RN}_{\mathcal{O}}, a, b, \in \mathcal{IN}_{\mathcal{O}}, \mathcal{O} \models (a, b) : r\}$. *The TBox part of the materialisation, also called TBox classification. is the set* $\{A \sqsubseteq B | A, B \in \mathcal{CN}_{\mathcal{O}}, \mathcal{O} \models A \sqsubseteq B\}$.

A materialisation of an ontology tells us what the extension of each of its concepts and roles is, and likewise which concepts in the ontology subsume each other. For example, given the following TBox $\mathcal{T}_1$ (in $\mathcal{ALC}$), we can infer that $Koala \sqsubseteq Herbivore$ is in its materialisation/TBox classification.

[3] An $\mathcal{SROIQ}$ concept is in NNF *iff* negation is applied only to atomic concepts, nominals or Self-restriction. NNF of a given concept can be computed in linear time[29].

Table 1: DL Semantics

Interpretation of concepts and roles		
$\top^{\mathcal{I}}$	$=$	$\Delta^{\mathcal{I}}$
$\bot^{\mathcal{I}}$	$=$	$\emptyset$
$\{a\}^{\mathcal{I}}$	$=$	$\{a^{\mathcal{I}}\}$
$(\neg C)^{\mathcal{I}}$	$=$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
$(C \sqcap D)^{\mathcal{I}}$	$=$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
$(\exists R.C)^{\mathcal{I}}$	$=$	$\{x \mid \exists y.\langle x, y\rangle \in R^{\mathcal{I}}, y \in C^{\mathcal{I}}\}$
$(\geq nS.C)^{\mathcal{I}}$	$=$	$\{x \mid \sharp\{y.\langle x, y\rangle \in S^{\mathcal{I}}, y \in C^{\mathcal{I}}\} \geq n\}$
$Inv(R)^{\mathcal{I}}$	$=$	$\{\langle x, y\rangle \mid \langle y, x\rangle \in R^{\mathcal{I}}\}$
$(\neg R)^{\mathcal{I}}$	$=$	$\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \setminus R^{\mathcal{I}}$
Satisfaction of axioms w.r.t. an interpretation		
$\mathcal{I} \models R_1 \sqsubseteq R_2$	iff	$R_1^{\mathcal{I}} \subseteq R_2^{\mathcal{I}}$
$\mathcal{I} \models R_1 \circ \cdots \circ R_n \sqsubseteq R_{n+1}$	iff	$R_1^{\mathcal{I}} \times \cdots \times R_n^{\mathcal{I}} \subseteq R_{n+1}^{\mathcal{I}}$
$\mathcal{I} \models Dis(S_1, S_2)$	iff	$S_1^{\mathcal{I}} \cap S_2^{\mathcal{I}} = \emptyset$
$\mathcal{I} \models C \sqsubseteq D$	iff	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
$\mathcal{I} \models a : C$	iff	$a^{\mathcal{I}} \in C^{\mathcal{I}}$
$\mathcal{I} \models (a, b) : R$	iff	$\langle a^{\mathcal{I}}, b^{\mathcal{I}}\rangle \in R^{\mathcal{I}}$
$\mathcal{I} \models (a, b) : \neg R$	iff	$\langle a^{\mathcal{I}}, b^{\mathcal{I}}\rangle \notin R^{\mathcal{I}}$
$\mathcal{I} \models a \doteq b$	iff	$a^{\mathcal{I}} = b^{\mathcal{I}}$
$\mathcal{I} \models a \neq b$	iff	$a^{\mathcal{I}} \neq b^{\mathcal{I}}$

Example 1 *An example TBox $\mathcal{T}_1$ includes the following axioms:*

$$Koala \sqsubseteq \forall eat.(\exists partof.Eucalypt) \tag{1}$$

$$Eucalypt \sqsubseteq Plant \tag{2}$$

$$Plant \sqcup \exists partof.Plant \sqsubseteq VegeFood \tag{3}$$

$$\forall eat.VegeFood \sqsubseteq Herbivore \tag{4}$$

3. Related Work and Motivations

In this section, we motivate our work on tractable approximate deduction by reviewing different existing approaches of ontology reasoning. As we discussed in the introduction, we categorise these approaches into fully-fledged universal solutions for expressive DLs, dedicated solutions for tractable DLs and solutions that exploit the gaps in between, such as approximation.

3.1. Reasoning in Expressive DLs

Traditionally in expressive DLs, reasoning is performed by tableau-based algorithms [17, 31, 32]. It constructs a tableau (as a witness of a model of ontology) as a graph in which each node x represents an individual and is labelled with a set of concepts it must satisfy, each edge $\langle x, y\rangle$ represents a pair of individuals satisfying all roles that label the edge. Subsumption checking $\Sigma \models C \sqsubseteq D$ can be reduced to consistency checking of an extended knowledge base $\Sigma \cup \{x : C \sqcap \neg D\}$, where x is a fresh individual. To test this, a tableau is initialised with the concept and role assertions in Σ and a node x labelled with $C \sqcap \neg D$, and is then expanded by repeatedly applying the completion rules. To classify an ontology, tableau algorithms in general iterate all necessary pairs of concepts, and try to construct a model of the ontology that violates the subsumption relation between them [41]. Instance checking $\Sigma \models a : C$ is reduced in a similar way to consistency checking of $\Sigma \cup \{a : \neg C\}$ [69].

Due to the model construction mechanism of the tableau algorithms, GCIs become major sources of non-determinism for such algorithms and contribute to their intractability. For each GCI $C \sqsubseteq D$ in the ontology, a tableau algorithm will add a meta-constraint $\neg C \sqcup D$ into the label of each node of the tableau. The algorithm then first extends with one of the disjuncts $\neg C$ (or D). If it finds a clash, it backtracks and extends with the other disjunct D (or $\neg C$). If there are n GCIs, in worst case this will lead to 2^n expansions for each node of the tableau. This exponentially enlarges the search space.

Some techniques have been developed to deal with GCIs. *Absorption* [80] can reduce a GCI, e.g. $A \sqcap C \sqsubseteq D$ with A being an atomic concept, into a non-GCI, e.g., $A \sqsubseteq \neg C \sqcup D$; however, it is only applicable for GCIs whose LHS is a conjunction with an atomic concept as a conjunct or whose RHS is a complement of atomic concept or a disjunction with a complement of atomic concept as a disjunct. *(Extended) Role Absorption* [79, Sec.4.1] can absorb GCIs of form $\exists r.C \sqsubseteq D$ ($C \sqsubseteq \forall r.D$) into domain (range) constrains. For example axiom (3) can be decomposed into $\exists partof.Plant \sqsubseteq VegeFood$ and thus be absorbed as $\exists partof.\top \sqsubseteq VegeFood \sqcup \neg\exists partof.Plant$, which means that the domain of $partof$ is $VegeFood \sqcup \neg\exists partof.Plant$. However, the applicability of this approach is still limited and it still contains a disjunction in the domain. *Binary Absorption* [36] tries to rewrite GCIs into form $A_1 \sqcap A_2 \sqsubseteq C$ where A_1 and A_2 are named concepts. To sum up, the above absorptions can only be applied to a limited patterns of GCIs; e.g., axiom (4) can not be dealt with by any absorption optimisation.

Motik et al. [59] proposed a novel reasoning calculus for very expressive DLs by combining *hypertableau and hyperresolution* technologies, which is later implemented in the HermiT reasoner. This calculus generalises all known absorption variants by transforming DL axioms into DL-clauses while maintaining their Hornness as much as possible. This approach can eliminate all unnecessarily non-determinism introduced by Horn-GCIs hence it is deterministic and very efficient for Horn-TBox. However, similar as other absorption technologies, it can not deal with non-Horn GCIs, such as the axiom (4) in our example. Even for a pure Horn-TBoxes, its reasoning efficiency is restricted by the complexity of the corresponding Horn-DLs, which in most cases is the same as their non-Horn siblings [46].

Sirin et al. [74] claimed that the appearance of nominals will introduce new challenges to tableau algorithm and presented a suite of techniques for optimising tableau algorithm in the presence of nominals. These techniques extensively exploited the use of "OneOf" (i.e. $\{a, b, c, \dots\}$) and "hasValue" (i.e. $\exists R.\{a\}$) constructs in ontologies to absorb GCIs with nominals. For example, axiom $\{a_1, a_2, \dots, a_n\} \sqsubseteq C$ can be equivalently replaced by a set of axioms $a_1 : C$, $a_2 : C$, $\dots$, $a_n : C$. However, such an optimisation (together with many other optimisations presented by Sirin [74]) can not cover GCIs of many other patterns. For example, from $A \equiv \forall r.\{a, b\}, B \equiv \forall r.\{a\}$ it will be intuitive to infer that $B \sqsubseteq A$ but the two GCIs can not be absorbed by any existing absorption technique. Even with optimisations, state-of-the-art fully-fledged tableau and hypertableau reasoners still cannot provide "responsive" reasoning services for some (smallish) ontologies with complex GCIs and nominals, such as the WINE Ontology.

3.2. Reasoning in Tractable DLs

Tractable DLs such as the DL-Lite family [11] and $\mathcal{EL}$ family [1] enjoy a much lower complexity than expressive DLs. In this paper, we are mainly interested in $\mathcal{EL}^{++}$ reasoning. Baader et al. [1] presented a set of TBox completion rules (Table 2) [4] to compute, given a normalised $\mathcal{EL}^{++}$ TBox $\mathcal{T}$, the TBox classification of $\mathcal{T}$. The reasoning is initialised with $\mathcal{T} \cup \bigcup_{A \in \mathcal{CN}_\mathcal{T}} \{A \sqsubseteq A, A \sqsubseteq \top\}$, and performed by repeatedly applying **R1-8** until no more rule can be applied. Such a procedure is tractable [1] and sound. Kazakov et al. [43] later showed that this procedure is also complete for nominal-safe $\mathcal{EL}^{++}$ ontologies. In this case, **R6** will only be applicable to inference of subsumptions between singleton nominals, i.e. if $\{x\} \sqsubseteq \{a\}$ and $\{y\} \sqsubseteq \{a\}$, then $\{x\} \sqsubseteq \{y\}$. Note that this implies $\{x\} \sqsubseteq \{y\}$ if $\{y\} \sqsubseteq \{x\}$. In this paper we will extend on this procedure. When an ABox $\mathcal{A}$ presents, an additional concept $C_\mathcal{A} := \bigsqcap_{C(a) \in \mathcal{A}} \exists u.(\{a\} \sqcap C) \sqcap \bigsqcap_{r(a,b) \in \mathcal{A}} \exists u.(\{a\} \sqcap \exists r.\{b\})$, where u is a fresh role name, is introduced. To this end, instance checking $a : C$ can be reduced to subsumption checking $\{a\} \sqcap C_\mathcal{A} \sqsubseteq C$, which can be realised by **R1-R8**. In general, such a reasoning approach that computes the closure of a set of completion rules on a set of axioms is called a *consequence-based* (or *consequence-driven*, *completion-based*, *saturation-based* by different authors) procedure.

Table 2: $\mathcal{EL}^{++}$ TBox completion rules (no datatypes)

R1	If $X \sqsubseteq A$ and $A \sqsubseteq B$ then $X \sqsubseteq B$
R2	If $X \sqsubseteq A_1, \ldots, X \sqsubseteq A_n$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq B$ then $X \sqsubseteq B$
R3	If $X \sqsubseteq A$ and $A \sqsubseteq \exists r.B$ then $X \sqsubseteq \exists r.B$
R4	If $X \sqsubseteq \exists r.A$, $A \sqsubseteq A'$ and $\exists r.A' \sqsubseteq B$ then $X \sqsubseteq B$
R5	If $X \sqsubseteq \exists r.A$ and $A \sqsubseteq \bot$ then $X \sqsubseteq \bot$
R6	If $X, A \sqsubseteq \{a\}$ and $X \leadsto_R A$ then $X \sqsubseteq A$
R7	If $X \sqsubseteq \exists r.A$ and $r \sqsubseteq s$ then $X \sqsubseteq \exists s.A$
R8	If $X \sqsubseteq \exists r_1.A$, $A \sqsubseteq \exists r_2.B$ and $r_1 \circ r_2 \sqsubseteq r_3$, then $X \sqsubseteq \exists r_3.B$

Kazakov et. al. [42] showed that a consequence-based reasoning procedure of $\mathcal{EL}$, a slightly weaker DL, can be parallelised with carefully designed completion rules. Such a parallel algorithm has been implemented in the ELK reasoner and demonstrated high performance on some of the most difficult $\mathcal{EL}$ ontologies. Krötzsch also showed that a consequence-based procedure can be developed for classification of other tractable DLs such as OWL 2 RL [45].

More importantly, by revising and extending the $\mathcal{EL}$ completion rules, Kazakov showed that a very efficient consequence-based TBox classification procedure can be developed for intractable DL Horn-$\mathcal{SHIQ}$ [41]. This procedure is further extended to support non-Horn DL $\mathcal{ALCH}$ [73].

Nevertheless, the consequence-based algorithms are restricted to limited expressive power when applied on the $\mathcal{EL}$ family or OWL 2 RL, and lose their tractability when applied on more expressive but intractable DLs such as Horn-$\mathcal{SHIQ}$ and $\mathcal{ALCH}$.

Dedicated $\mathcal{EL}$ algorithms can also be combined with intractable algorithms to provide better performance on ontologies beyond $\mathcal{EL}$. A recent work by Romero et. al [67] tries to combine the strength of a fully-fledged $\mathcal{SROIQ}$ reasoner and a very efficient $\mathcal{EL}$ reasoner in TBox classification by applying modularisation techniques. Particularly, the atomic concepts of an ontology

[4] In **R6** $X \leadsto_R A$ iff there exists $C_1, \ldots, C_k \in \mathcal{CN}_\mathcal{T}$ $(k \geq 1)$ s.t. $C_1 = X$ or $C_1 = \{b\}$, $C_j \sqsubseteq \exists r_j.C_{j+1}$ for some $r_j \in \mathcal{RN}_\mathcal{T}(1 \leq j < k)$ and $C_k = A$.

is partitioned into two sets, one's classification can be fully realised by only some $\mathcal{EL}$ axioms in the ontology, the other's can only be fully realised by involving some non-$\mathcal{EL}$ axioms in the ontology. Then a $\mathcal{SROIQ}$ reasoner such as HermiT is used to classify the non-$\mathcal{EL}$ signature, and an $\mathcal{EL}$ reasoner such as ELK is used to classify the $\mathcal{EL}$ signature. This approach systematically analyses and exploits the phenomena that not all axioms in an ontology are equally complex. The result is sound, complete and shown rather efficient in evaluation.

Nevertheless, it has a limitation that it does not always guarantee reduction of complexity. Hence its applicability is restricted to ontologies with large percentages of $\mathcal{EL}$ axioms. When only a small number of the concepts can be fully classified by $\mathcal{EL}$ axioms alone, its effect on improving reasoning efficiency might not be significant. This can be seen when the ontology axioms are quite complex or are tightly coupled. The FMA ontology in our evaluation is an example.

3.3. Reasoning with Approximation-based Approaches

The trade-off between representation expressiveness and computational efficiency has been acknowledged in research of Artificial Intelligence and Knowledge Representation for decades [50]. In order to address this trade-off, various early works on approximate reasoning of propositional and first-order logic have been developed. Horvitz investigated the use of incomplete inference mechanism by limiting the inference steps in resource-bounded environment [34, 35]. Levesque [47, 48, 49], Frisch [22, 23] and Patel-Schneider [64] explored *approximation deduction* with non-traditional semantics, such as multi-valued logics, or weaker notion of inferences that disallow complex inference steps. Another important research track follows the spirit of anytime reasoners [5], in which, it provides partial answers if it is stopped anytime during the computation, but the completeness improves with the time used in computing the answer. Suck kind of anytime reasoning is also suggested by Kautz [38] in the algorithm of knowledge compiling. Dalal [14, 15] firstly designed a approximation family for clausal logic, in which each reasoner can be decided in polynomial time. An anytime family of reasoners is a sequence $\vdash_0$, $\vdash_1$,... of reasoners such that each reasoner is tractable, each $\vdash_{i+1}$ is at least as complete as $\vdash_i$, and for each theory Π, there is a $\vdash_k$ complete for reasoning with Π. This polynomial approach was further extended by Finger [19] to full classical logic.

Kautz and Selman [38, 39, 71, 72] proposed the *knowledge compilation* approach. This approach essentially consists of two stages: In the off-line stage, it compiles a knowledge base KB in a more expressive logic into two knowledge bases KB_U and KB_L in less expressive logics, such that $KB_L \models KB \models KB_U$. KB_L is called the lower bound and KB_U is called the upper bound. In the on-line stage, if KB_L and KB_U yield the same results for a reasoning problem, then KB should also yield the same result. Since KB_L and KB_U are easier to reason with than KB, such results can be computed more efficiently than on KB. If the upper bound and lower bound yield different results, one can always use a fully-fledged reasoner to perform reasoning in KB. For more details of knowledge compilation, we refer our readers to the survey written by Cadoli and Donini [7].

Cadoli and Schaerf furthered the idea of approximate deduction and proposed the *approximate entailment* approach [8, 70, 9]. They model a reasoning problem as deciding whether a string x belongs to a set D or not. Their method then generates approximate sets $D_0, \ldots, D_m$ and $D^0, \ldots, D^n$ such that $D_0 \subseteq \cdots \subseteq D_m = D = D^n \subseteq \cdots \subseteq D^0$, where n and m are polynomially bounded by x. Then, x belongs to D if it belongs to any D_i, and x does not belong to D if it does not belong to some D^j. D_i (D^j) is generated in a way that deciding membership in D_0 (D^0) is a polynomial problem and gets exponentially harder as i grows but is no harder than deciding membership in D. Hence, the original reasoning problem can be computed more efficiently by proving (disproving) membership in $D_0, \ldots, D_m$ ($D^0, \ldots, D^n$). Their approach was later extended by Massacci [53], Finger and Wassermann [21, 20].

The ideas of many propositional and first-order approximate reasoning approaches have been adopted by DL researchers. Many approximation approaches [61, 16, 51, 6] combine the idea of knowledge compilation and language weakening by first compiling an ontology in a more complex DL L_1 into one (or two) ontology(ies) in a less complex formalism L_2, and then using a L_2 reasoner to reason with the compiled ontology(ies), which can be regarded as approximation(s) of the original ontology. For example, the *semantic approximation* technology [61] (also referred to as global semantic approximations [13]) uses a fully-fledged DL reasoner to compute all entailments of an ontology that can be expressed with a tractable DL, such as DL-Lite. The computed entailments are compiled into a DL-Lite ontology and can be queried against with a DL-Lite reasoner in run-time. When both the compilation and L_2 reasoning are tractable, the entire approximate reasoning procedure is tractable.

Other approximation approaches [25, 76, 83, 28, 81] apply the idea of approximate deduction by simplifying the treatment of complex axioms, expressions or constructors in ontology reasoning. Such a simplification can be realised by replacing a complex concept expression with a simpler one in the ontology [25] or query [76] or both [83], or by replacing a harder-to-deal-with constructor, such as disjunction that leads to non-determinism, with a easier-to-deal-with constructor, such as conjunction [28, 81]. Either, the deduction procedures are simplified as they do not need to operate w.r.t. the full semantics of the original problem. For example, Groot et al. [25] adopted the approximate entailment idea to speed up concept unsatisfiability checking via approximation. Given a concept C, it constructs a sequence of $C_i^\top$ such that $C \sqsubseteq \cdots \sqsubseteq C_1^\top \sqsubseteq C_0^\top$, and a sequence of $C_i^\perp$ such that $C_0^\perp \sqsubseteq C_1^\perp \sqsubseteq \ldots C$ by replacing all existential restrictions ($\exists R.D$) after i universal quantifiers ($\forall$) inside C with $\top$ and $\perp$ respectively.Then C is unsatisfiable (satisfiable) if some $C_i^\top$ ($C_i^\perp$) is unsatisfiable (satisfiable). Since $C_i^\top$ ($C_i^\perp$) is usually simpler than C, its (un)satisfiability checking should also be easier.

The hotspot-based approximation [24] applies a similar idea as the modularisation-based reasoning. It identifies a small subset of the ontology, called a hotspot, whose removal will significantly reduce the classification time of the rest of the ontology, and then removes this hotspot to boost reasoning.

Despite the successful application of approximation technologies, most DL approximation approaches still have one of the following limitations:

1. **No substantial reduction of complexity:** For example, Groot et al.'s approximate deduction approach [25] simplifies the unsatisfiability of a concept expression by replacing the existential restrictions in the test concept with $\top$ and $\perp$. However, this approach only approximates the tested concept, but not the ontology. Also, When the test concept subsumption contains no existential restriction, such as $Koala \sqsubseteq Herbivore$, this approach doesn't work. Thus the unsatisfiability checking still requires fully-fledged reasoning of the original ontology. In other words, it does not necessarily reduce the complexity of reasoning.
2. **Limited expressive power:** For example, Hitzler et al.'s resolution-based approximation [28] uses the KAON2 algorithm to perform reasoning. Similarly, the `SCREECH` [28, 81] approach utilises the KAON2 algorithm to translate a $\mathcal{SHIQ}$ TBox into disjunctive datalog, and execute the rules together with a $\mathcal{SHIQ}$ ABox and a query by a datalog engine. By rewriting or eliminating all the disjunctive rules the data complexity can be

reduced from coNP-complete of OWL DL to polynomial time. Both of them rely on the specific $\mathcal{SHIQ}$ reasoner KAON2 [55]. The SHER system [16] summaries the ABox to provide more efficient instance retrieval for the DL $\mathcal{SHIN}$, a sub-language of $\mathcal{SHIQ}$. The DLog system [51] applies Prolog to support efficient $\mathcal{SHIQ}$ reasoning. Because $\mathcal{SHIQ}$ does not support role chain or nominal, the applicability of these approaches are affected.

3. **Non-trivial off-line compilation:** Semantic Approximation [61, 6] directly applies knowledge compilation. Particularly, Pan and Thomas' approach [61] approximates to DL-Lite$_F$ and is later extended by Botoeva et al. to approximate to DL-Lite$_A$ [6] by introducing fresh names to represent complex expressions. Both approaches can produce optimal approximations in terms of soundness and completeness of query results. However such a guarantee of quality is expensive. They both use a fully-fledged reasoner to materialise the ontology and store in a database to speed up on-line query answering. Although they supports arbitrary language, they need time-consuming preprocessing. The hotspot-based approximation [24] discovers hotspots by comparing the effect of removing different candidates, which itself requires a lot of time, and in many cases making the whole procedure taking even longer time than classifying the ontology directly. Therefore they are not efficient for on-line applications.

To sum up, fully-fledged reasoning algorithms (Sec. 3.1) have difficulties to handle complex structured axioms; tractable DL algorithms and modularisation-based approach (Sec. 3.2) can not support more expressive languages without losing tractability; traditional approximation approaches (Sec. 3.3) do not always reduce reasoning complexity and can be limited on expressive power and/or efficiency. In what follows, we present our approach which is motivated and inspired by these works, and show that it works well with real-world and benchmark ontologies.

4. Approach Overview

Before presenting the formal technical details, in this section we outline our approach with examples and a brief overview . The basic idea behind our approach is that expressive constructors used in ontologies do not always contribute to reasoning in a necessarily complex manner, hence expressions involving them can be approximated with simpler ones and be exploited with simpler inference procedures.

Considering the $\mathcal{T}_1$ in Example 1, it is easy to derive the following axiom:

$$\exists partof.Eucalypt \sqsubseteq VegeFood.$$

Given this, it is sufficient to derive the following axiom, in which we use underline to highlight the expressions appearing in the above axiom:

$$\forall eat.\underline{(\exists partof.Eucalypt)} \sqsubseteq \forall eat.\underline{VegeFood}.$$

Note that this can be achieved by a tractable inference rule:

$$C \sqsubseteq D \rightarrow \forall r.C \sqsubseteq \forall r.D$$

or the combination of two tractable inference rules:

$$C \sqsubseteq D \rightarrow \neg D \sqsubseteq \neg C$$
$$C \sqsubseteq D \rightarrow \exists r.C \sqsubseteq \exists r.D$$

Although there are other more complex axioms in $\mathcal{T}_1$, such as axiom (4), they are not necessarily required in this step of inference. From this point, we can further combine with axiom (1) and (4) to perform the following inference, in which we also use underline to highlight the sub- and super-expressions from the previous axiom:

$$Koala \sqsubseteq \underline{\forall eat.\exists partof.Eucalypt} \sqsubseteq \underline{\forall eat.VegeFood} \sqsubseteq Herbivore$$

As we can see, although both $\forall eat.\exists partof.Eucalypt$ and $\forall eat.VegeFood$ contains expressive constructors, these constructors do not contribute to the inference of $Koala \sqsubseteq Herbivore$. Hence the two complex expressions can be simply treated as named concepts in this step of inference.

This procedure can be applied on ABox reasoning as well. Most straight-forwardly, one can always internalise ABox into TBox and use the same TBox reasoning procedure to materialise ABox. Or, when the ontology contains no nominal, such as the $\mathcal{T}_1$ in the Koala example, one can always classify TBox first and then use a more specialised and optimised version of TBox reasoning procedure on ABox. When the ontology contains nominal only in a restricted manner, such as under the nominal-safety condition, further optimisation can be made.

To summarise the above findings, we can perform tractable reasoning with complex ontologies by:

1. treating complex expressions as names because the expressive constructors used within do not necessarily contribute to reasoning;
2. applying tractable inference procedure of a simpler DL since the above procedure have simplified the syntax of ontology. For example, in the above inference we are primarily using $\mathcal{EL}^{++}$ inference;
3. extending inference procedure with additional tractable rules to partially capture the semantics beyond the simpler DL. The $C \sqsubseteq D \rightarrow \neg D \sqsubseteq \neg C$ above is an example;
4. extending formalisms to support the above additional rules. For example $\mathcal{EL}^{++}$ does not support complement. To apply the $C \sqsubseteq D \rightarrow \neg D \sqsubseteq \neg C$, we need to maintain the complementary relations between concepts;
5. optimising inference rules to support ABox with TBox containing nominals.

The above 5 points constitute the major components of our approach. Particularly, points 1 and 4 are formalised with the transformations we will present later, which will be used to support the $\mathcal{EL}^{++}$ inference rules and additional tractable rules. Points 2, 3 and 5 are formalised with the approximate deduction. For ontologies satisfying certain syntactic or semantic criteria, more optimised ABox rules can be applied. In the next two sections, we will present the formalisations and their characteristics.

5. Syntactic Approximation for TBox Reasoning

In this section, we present syntactic approximation as an approach to tractable approximate reasoning for TBox. We introduce the definition of approximation and completion rules. In later sections, they will be extended to support ABox reasoning.

In approximation, we only consider concepts and roles (together they are called predicates) corresponding to the particular ontology in question. We use the notion *term* to refer to these “interesting” concept and role expressions:

Definition 2 **(Term)** *For an ontology $\mathcal{O}$, a concept or a role expression t is a term of $\mathcal{O}$ if*

1. *t appears in any axiom of $\mathcal{O}$, or*
2. *the complement of t is a term, or*
3. *the inverse of t is a term when t is a role expression, or*
4. *the syntactic sub-expression of t is a term when t is a concept expression.*

In situation 2, the complement of a role can be interpreted as shown in Table 1. In order to represent all these terms that will be used in $\mathcal{EL}^{++}$ reasoning, we first assign names to them.

Definition 3 ***(Name Assignment)*** *Given an ontology $\mathcal{O}$, a* name assignment $fa(\centerdot)$ *is a function such that for each concept expression C (each role expression R) constructed from $\Sigma_{\mathcal{O}}$, we have: $fa(C) = C$ if $C \in \mathcal{CN}_{\mathcal{O}}$, $fa(R) = R$ if $R \in \mathcal{RN}_{\mathcal{O}}$, otherwise $fa(C)$ or $fa(R)$ is a fresh name.*

For each ontology there can be infinite number of concepts and roles to assign names to. The following Table 3 illustrates name assignments of some complex terms in Example 1. They will be used in this section to illustrate the idea of syntactic approximation. These fresh names are chosen arbitrarily. It is easy to see that $\forall eat.\exists partof.Eucalypt$ is a complement of $\exists eat.\forall partof.\neg Eucalypt$, hence we assign C_1 and cC_1 to them, respectively, just to make the complementary relation clearer. Similar applies to other complementary terms.

Table 3: Name assignment

Term	Name
$\forall eat.\exists partof.Eucalypt$	C_1
$\exists eat.\forall partof.\neg Eucalypt$	cC_1
$\forall partof.\neg Eucalypt$	C_2
$\exists partof.Eucalypt$	cC_2
$Plant \sqcup \exists partof.Plant$	C_3
$\neg Plant \sqcap \forall partof.\neg Plant$	cC_3
$\forall partof.\neg Plant$	C_4
$\exists partof.Plant$	cC_4
$\forall eat.VegeFood$	C_5
$\exists eat.\neg VegeFood$	cC_5
$\neg Plant$	$cPlant$
$\neg VegeFood$	$cVegeFood$

5.1. TBox Transformation

Now we transform an $\mathcal{SROIQ}$ TBox to an $\mathcal{EL}^{++}$ TBox plus a complement table (CT), a cardinality table (QT) and a inverse table (IT). Elements of CT are pairs (A, B) where A and B are names assigned to a term and its complement, respectively. Elements of QT are 4-tuples (A, B, r, n) where A and B are term names, r a role name and n an integer number. Elements of IT are pairs (r, s) where r and s are names assigned to a role expression and its inverse. The basic idea is to represent (non-$\mathcal{EL}^{++}$) terms with their name assignments.

Definition 4 ***($\mathcal{EL}^{++}_{\mathcal{CQI}}$ TBox Transformation)*** *Given an ontology $\mathcal{O}$ and a name assignment $fa(\centerdot)$, its $\mathcal{EL}^{++}_{\mathcal{CQI}}$* TBox approximation $TA_{fa,\mathcal{EL}^{++}_{\mathcal{CQI}}}(\mathcal{O})$ *is a four-tuple $(\mathcal{T}, CT, QT, IT)$ constructed as follows:*

1. *$\mathcal{T}, CT, QT$ and IT are all initialised to $\emptyset$.*
2. *for each $C \sqsubseteq D$ ($C \equiv D$) in $\mathcal{O}$, $\mathcal{T} = \mathcal{T} \cup \{fa(C) \sqsubseteq fa(D)\}$ ($\mathcal{T} = \mathcal{T} \cup \{fa(C) \equiv fa(D)\}$).*
3. *for each RI axiom $\beta \in \mathcal{O}$, $\mathcal{T} = \mathcal{T} \cup \{\beta'\}$, where β' is obtained by replacing each role R in β with $fa(R)$.*
4. *for each axiom $Dis(R, S) \in \mathcal{O}$, $\mathcal{T} = \mathcal{T} \cup \{fa(R) \sqsubseteq fa(\neg S)\}$*
5. *for each term C in $\mathcal{O}$, $CT = CT \cup \{(fa(C), fa(\sim C))\}$, and if C is of the form:*
   (a) *$C_1 \sqcap \cdots \sqcap C_n$, then $\mathcal{T} = \mathcal{T} \cup \{fa(C) \equiv fa(C_1) \sqcap \cdots \sqcap fa(C_n)\}$,*
   (b) *$\exists R.D$, then $\mathcal{T} = \mathcal{T} \cup \{fa(C) \equiv \exists fa(R).fa(D)\}$*
   (c) *$\geq nR.D$, then*
      i. *if $n = 0$, $\mathcal{T} = \mathcal{T} \cup \{\top \sqsubseteq fa(C)\}$*
      ii. *if $n = 1$, $\mathcal{T} = \mathcal{T} \cup \{fa(C) \equiv \exists fa(R).fa(D)\}$*
      iii. *otherwise, $\mathcal{T} = \mathcal{T} \cup \{fa(C) \sqsubseteq \exists fa(R).fa(D)\}$ and $QT = QT \cup \{(fa(C), fa(D), fa(R), n)\}$.*
6. *for each $r \in RN_{\mathcal{O}}$, $IT = IT \cup \{(fa(r), fa(Inv(r))), (fa(Inv(r)), fa(r)), (fa(\neg r), fa(\neg(Inv(r)))), (fa(\neg(Inv(r))), fa(\neg r))\}$.*

The $\mathcal{EL}^{++}_{\mathcal{CQI}}$ in definition indicates that the transformation extends $\mathcal{EL}^{++}$ with additional information maintaining complements ($\mathcal{C}$), cardinality restrictions ($\mathcal{Q}$) and inverse roles ($\mathcal{I}$), and is an extension of the $\mathcal{EL}^{++}_{\mathcal{C}}$, $\mathcal{EL}^{++}_{\mathcal{CQ}}$ transformations presented in our earlier work [65]. In the rest of the paper, we will omit the $\mathcal{EL}^{++}_{\mathcal{CQI}}$ to simplify the notation since in this paper all the work are based on only the $\mathcal{EL}^{++}_{\mathcal{CQI}}$ transformation. Hence we will simply call *TBox transformation* and denote it with $TA_{fa}(\mathcal{O})$.

As we mentioned in the overview section, the transformation and its components will be used in reasoning. For convenience, we also define a *complement function* $fc(\centerdot)$ as: for each concept (including nominal) or role name A such that $(A, B) \in CT$, $fc(A) = B$ and $fc(B) = A$. And we define a *inverse name function* $fi(\centerdot)$ as: for each role name r with $(r, s) \in IT$, we have $fi(r) = s$ and $fi(s) = r$. Since both CT and IT contain only symmetric pairs, $fc(\centerdot)$ and $fi(\centerdot)$ are well-defined.

This Def. 4 deserves some explanations:

- Step-2 rewrites all the CIs;
- Step-3 rewrites all the RIs;
- Step-4 defines the relations between disjoint roles;
- Step-5 approximates terms and constructs the complement table CT and the cardinality table QT. Non-$\mathcal{EL}^{++}$ terms such as disjunctions will be approximated via their complements. For example, $\leq nR.D$ will be approximated via the approximation of its complement $\geq (n+1)R.D$. Furthermore, this step also constructs the subsumption between a cardinality restriction and an existential restriction. Particularly, if $i \geq 2$, then $\geq ir.A \sqsubseteq \exists r.A$;
- Step-6 maintains the inverse relations.

Based to Table 3, we can transform the TBox $\mathcal{T}_1$ into $(\mathcal{T}_{Koala}, CT_{Koala}, QT_{Koala}, IT_{Koala})$ as follows:

Example 2 *$\mathcal{T}_{Koala}$ contains axioms generated by Step 2 and 8. Particularly, axiom (1) $Koala \sqsubseteq \forall eat.(\exists partof.Eucalypt)$ is transformed into:*

$$Koala \sqsubseteq C_1 \tag{5}$$
$$cC_1 \equiv \exists eat.C_2 \tag{6}$$
$$cC_2 \equiv \exists partof.Eucalypt \tag{7}$$

Axiom (2) $Eucalypt \sqsubseteq Plant$ is preserved. Axiom (3) $Plant \sqsubseteq \exists partof.Plant \sqsubseteq VegeFood$ is transformed into:

$$C_3 \sqsubseteq VegeFood \tag{8}$$
$$cC_3 \equiv cPlant \sqcap C_4 \tag{9}$$
$$cC_4 \equiv \exists partof.Plant \tag{10}$$

Axiom (4) $\forall eat.VegeFood \sqsubseteq Herbivore$ is transformed into:

$$C_5 \sqsubseteq Herbivore \tag{11}$$
$$cC_5 \equiv \exists eat.cVegeFood \tag{12}$$

CT_{Koala} contains pairs such as (C_1, cC_1), (C_2, cC_2), (C_3, cC_3), (C_4, cC_4), (C_5, cC_5), $(Plant, cPlant)$, $(VegeFood, cVegeFood)$, etc.

It is easy to show that the TBox transformation can be performed in linear time since it monotonically generates linear-sized results:

Proposition 1 ***(TBox Linear Transformation)*** *In Def. 4, transforming from $\mathcal{O}$ to $TA_{fa}(\mathcal{O})$ can be done in linear time.*

This proposition can be easily proved by investigating the origins of different axioms in the approximation (cf. Appendix).

The transformation results can also be regarded as a syntactic variants of the original ontology that preserves TBox classification:

Theorem 2 ***(TBox Reasoning Preservation)*** *For any ontology $\mathcal{O} = (\mathcal{T}_\mathcal{O}, \mathcal{A}_\mathcal{O})$ and its TBox transformation $(\mathcal{T}, CT, QT, IT)$, let $\mathcal{T}'$ be a TBox constructed as follows:*

1. *$\mathcal{T}'$ is initialised as $\mathcal{T}$;*
2. *for any concept pair $(A, B) \in CT$, $\mathcal{T}' = \mathcal{T}' \cup \{A \equiv \neg B\}$;*
3. *for any role pair $(r, s) \in CT$, $\mathcal{T}' = \mathcal{T}' \cup \{Dis(r, s)\}$;*
4. *for any $(A, B, r, n) \in QT$, $\mathcal{T}' = \mathcal{T}' \cup \{A \equiv \geq nr.B\}$;*
5. *for any $(r, s) \in IT$, $\mathcal{T}' = \mathcal{T}' \cup \{r \equiv Inv(s)\}$;*

then for any GCI α with $\Sigma_\alpha \subseteq \Sigma_{\mathcal{T}_\mathcal{O}}$, we have $\mathcal{T}_\mathcal{O} \models \alpha$ iff $\mathcal{T}' \models \alpha$:

This theorem can be proved by showing that $\mathcal{T}_\mathcal{O} \cup \mathcal{T}' \cup \mathcal{T}_{fa}$ is a $\Sigma_{\mathcal{T}_\mathcal{O}}$-conservative extension [52] of both $\mathcal{T}_\mathcal{O}$ and $\mathcal{T}'$, where $\mathcal{T}_{fa}$ is the set of definition $fa(P) \equiv P$ for all term P of $\mathcal{O}$ except complement of roles. Note that a TBox T_1 is an Σ-conservative extension of another TBox T_2 if $T_2 \subseteq T_1$, $Sigma \subseteq Sigma_{T_2}$ and for each GCI α with $\Sigma_\alpha \subseteq \Sigma$, we have $T_1 \models \alpha$

implies $T_2 \models \alpha$. Recall that $\Sigma_{\mathcal{T}}$ is the signature of $\mathcal{T}$ and Σ_α is the signature of $\{\alpha\}$. Hence, $\mathcal{T}_\mathcal{O} \models \alpha$ iff $\mathcal{T}_\mathcal{O} \cup \mathcal{T}' \cup \mathcal{T}_{fa} \models \alpha$ iff $\mathcal{T}' \models \alpha$. The full proof is deferred to the appendix.

An important implication of this theorem is that the TBox transformation itself can be used to fully preserve the semantics of the original TBox, if the information in CT, QT, IT can be fully exploited, while the $\mathcal{T}$ alone is a naive soundness preserving $\mathcal{EL}^{++}$ approximation of the original ontology.

Proposition 3 ***(TBox Approximation)*** *For an ontology $\mathcal{O}$, let $TA_{fa}(\mathcal{O})$ be $(\mathcal{T}, CT, QT, IT)$ as defined in Def. 4, then $\mathcal{T}$ is an $\mathcal{EL}^{++}$ TBox such that for any axiom α with $\Sigma_\alpha \subseteq \Sigma_\mathcal{O}$, we have $\mathcal{T} \models \alpha$ only if $\mathcal{O} \models \alpha$.*

As we can see, there is a gap between $\mathcal{T}$ and $\mathcal{T}'$. Such a gap can be bridged as much as possible by utilising CT, QT and IT. To achieve this while preserving the tractability of reasoning is a major motivation behind the rules we will introduce later in this section. In Sec. 8.2 we will show that there are certain gaps that can hardly be bridged in a tractable manner, notably resolution, cardinality counting and interactions between existential restrictions and inverse roles.

In order to apply the $\mathcal{EL}^{++}$ reasoning rules, we further perform normalisation of $\mathcal{T}$ by rewriting axioms of form $C \sqsubseteq D_1 \sqcap \cdots \sqcap D_n$ into $C \sqsubseteq D_1, \ldots, C \sqsubseteq D_n$. Since we only consider RIAs with no more than 3 roles, we do not need to further normalise RIAs. The normalisation can be done in linear time (this is the same case as in $\mathcal{EL}^+$ [4]). Apparently such rewriting does not change the semantics or vocabulary of $\mathcal{T}$ and the CT, QT and IT will not be changed. In the following, we assume $\mathcal{T}$ to be always normalised.

5.2. TBox Approximate Deduction Rules

Now we can do TBox approximate reasoning by utilising the complementary relations in CT, the cardinality information in QT and the inverse relations in IT. In addition to the original $\mathcal{EL}^{++}$ rules **R1-8** in Table 2, we devise some additional rules. Particularly, the CT information can be exploited by rules in Table 4. Note that the $fc(\textbf{.})$ is the complement function we have introduced after Def. 4.

Table 4: TBox completion rules for CT

R9	If $X \sqsubseteq A, X \sqsubseteq B$ and $A = fc(B)$ then $X \sqsubseteq \bot$
R10	If $B \sqsubseteq A$ then $fc(A) \sqsubseteq fc(B)$
R11	If $X \sqsubseteq A_1, \ldots, X \sqsubseteq A_{i-1}, X \sqsubseteq A_{i+1}, \ldots, X \sqsubseteq A_n$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq \bot$, then $X \sqsubseteq fc(A_i)$
R12	If $A \sqsubseteq \exists r.\{b\}$ and $B \sqsubseteq \exists fc(r).\{b\}$, then $A \sqsubseteq fc(B)$

R9 realises axiom $A \sqcap \sim A \sqsubseteq \bot$. **R10** realises $A \sqsubseteq B \rightarrow \sim A \sqsubseteq \sim B$. **R11** builds up the relations between conjuncts of a conjunction, *e.g.* $A \sqcap B \sqsubseteq \bot$ implies $A \sqsubseteq \sim B$. **R12** deals with disjoint roles, i.e. disjoint roles should not share any instance. Thus if instances of A and B are related to the same individual with disjoint relations, then A and B should also be disjoint with each other.

Now we can infer $Koala \sqsubseteq Herbivore$ from Example 2. Particularly, from (2) we can infer that $cC_2 \sqsubseteq cC_4$. With **R10** we can further infer that $C_4 \sqsubseteq C_2$. Hence combining with (9) we can infer the following:

$$cC_3 \sqsubseteq C_2 \tag{13}$$

Similarly, applying **R10** on (8) we can infer the following:

$$cVegeFood \sqsubseteq cC_3 \tag{14}$$

Combining (13) and (14) we can derive $cVegeFood \sqsubseteq C_2$, which further leads to $cC_5 \sqsubseteq cC_1$. Applying **R10** again we will have $C_1 \sqsubseteq C_5$ and finally $Koala \sqsubseteq Herbivore$ can be derived.

Now we consider a new example, in which cardinality restrictions are used:

Example 3 *A TBox $\mathcal{T}_3$ contains the following axioms:*

$$Primate \sqsubseteq \geq 4hasLeg.Leg \tag{15}$$

$$Human \sqsubseteq \leq 2hasLeg.Leg \tag{16}$$

$$Human \sqsubseteq Primate \tag{17}$$

Apparently, from this TBox we have $Human \sqsubseteq \bot$. This is an intuitive result due to the inconsistent understanding of Leg, i.e. in (15) Leg actually means limb *while in (16) Leg means* lower limb.

Assuming $TA_{fa}(\mathcal{T}_3) = (\mathcal{T}_{Primate}, CT_{Primate}, QT_{Primate}, IT_{Primate})$. Apparently, after transformation we should be able to obtain the following axioms from $T_{Primate}$:

$$Primate \sqsubseteq C_6 \tag{18}$$

$$Human \sqsubseteq C_7 \tag{19}$$

where $C_6 = fa(\geq 4hasLeg.Leg)$ and $C_7 = fa(\leq 2hasLeg.Leg)$. We also have $(C_7, cC_7) \in CT_{Primate}$ and $(C_6, Leg, hasLeg, 4), (cC_7, Leg, hasLeg, 3) \in QT_{Primate}$, where $cC_7 = fa(\geq 3hasLeg.Leg)$.

To derive the unsatisfiability of $Human$ we exploit the QT information with the rule in Table 5.

Table 5: TBox completion rules for QT

R13	If $A \sqsubseteq B$, $(X, A, r, i), (Y, B, s, j) \in QT$, $r \sqsubseteq s$ and $i \geq j$, then $X \sqsubseteq Y$

R13 realises inference $A \sqsubseteq B, R \sqsubseteq S, i \geq j \rightarrow \geq i\ R.A \sqsubseteq \geq j\ S.B$. Note that this **R13** is equivalent to the **R12** in our previous work [65]. While the **R13-16** in our previous work are omitted in this paper because they can be realised by the combination of the above **R13**, step 5.(c).iii of Def. 4 and the rules **R1**, **R4** and/or **R8**.

With this additional rule $Human \sqsubseteq \bot$ can be derived. First of all, applying **R13** we should have $C_6 \sqsubseteq cC_7$. This leads to the result that $Primate \sqsubseteq cC_7$ and hence $Human \sqsubseteq cC_7$. Together with (19) we apply rule **R9** to derive that $Human \sqsubseteq \bot$.

Due to the involvement of disjoint roles, inverse roles and nominals in axioms, the RIAs in the above rules becomes non-trivial. Therefore we propose the following additional rules (Table 6) to compute the RIAs and exploit the IT. For each role r, we initialise $r \sqsubseteq r$. Note that the $fi(\centerdot)$ is the inverse function we have introduced after Def. 4.

R14 realises $R \sqsubseteq S, S \sqsubseteq T \rightarrow R \sqsubseteq T$. **R15** realises $R \sqsubseteq S \rightarrow R^- \sqsubseteq S^-$. **R16** realises $R \sqsubseteq S \rightarrow \neg S \sqsubseteq \neg R$. **R17** realises $R_1 \circ R_2 \sqsubseteq R_3 \rightarrow R_2^- \circ R_1^- \sqsubseteq R_3^-$. Due to the role chain normalisation, this can be generalised to $R_1 \circ \cdots \circ R_n \sqsubseteq S \rightarrow R_n^- \circ \cdots \circ R_1^- \sqsubseteq S^-$.

Table 6: TBox completion rules for RIA and IT

R14	If $r \sqsubseteq s$ and $s \sqsubseteq t$ then $r \sqsubseteq t$
R15	If $r \sqsubseteq s$, then $fi(r) \sqsubseteq fi(s)$
R16	If $r \sqsubseteq s$, then $fc(s) \sqsubseteq fc(r)$
R17	If $r_1 \circ r_2 \sqsubseteq r_3$, then $fi(r_2) \circ fi(r_1) \sqsubseteq fi(r_3)$
R18	If $\exists r.\top \sqsubseteq X$, $X \sqsubseteq A$, $\exists fi(r).\top \sqsubseteq Y$, $Y \sqsubseteq B$, $\bigsqcap_{1 \leq i \leq n} fc(\{a_i\}) \sqsubseteq fc(A)$, $\bigsqcap_{1 \leq j \leq m} fc(\{b_j\}) \sqsubseteq fc(B)$, for any $1 \leq i \leq n$, $1 \leq j \leq m$, $fc(A) \sqsubseteq fc(\{a_i\})$, $fc(B) \sqsubseteq fc(\{b_j\})$ and $\{a_i\} \sqsubseteq \exists s.\{b_j\}$, then $r \sqsubseteq s$
R19	If $\{a\} \sqsubseteq \exists r.\{b\}$, then $\{b\} \sqsubseteq \exists fi(r).\{a\}$
R20	If $\{a\} \sqsubseteq \exists r.A$ and $A \sqsubseteq \exists fi(fc(r)).\{b\}$, then $\{a\} \sqsubseteq fc(\{b\})$

R18 needs more explanations: It realises a possible impact of nominal relations on role inclusions. According to DL semantics, $R \sqsubseteq S$ *iff* $R^{\mathcal{I}} \subseteq S^{\mathcal{I}}$ where $R^{\mathcal{I}}$ and $S^{\mathcal{I}}$ are the interpretations of R and S, respectively. Given a role R, the domain and range restrictions make up a super set of $R^{\mathcal{I}}$. Whereas ABox axioms regarding R make up a subset of $R^{\mathcal{I}}$. Therefore, suppose $\exists R.\top \sqsubseteq A$ and $\exists R^{-}.\top \sqsubseteq B$, if $a \in A, b \in B, (a, b) : S$ can be entailed, then $R \sqsubseteq S$ can be entailed. **R18** detects such a pattern by matching the transformed axioms.

Due to the above reason, nominal relations become non-trivial. **R19** realises the inverse relations between nominals. Although we are not considering ABox reasoning in this stage and only considering nominal-safe ontologies, such relation can still be important for TBox materialisation. An example is given in the following:

Example 4 *Given a TBox $\mathcal{T}_4$ containing the following axioms:*

$$
\begin{aligned}
&ScottishIsland \sqsubseteq \exists locatedIn.\{Scotland\},\\
&\{Britain\} \sqsubseteq \exists include.\{Scotland\},\\
&\exists partOf.\{Britain\} \sqsubseteq BritishPlace,\\
&locatedIn \circ include^{-} \sqsubseteq partOf,
\end{aligned}
$$

using **R19** to infer $\{Scotland\} \sqsubseteq \exists fi(include).\{Britain\}$ is significant for the derivation of $ScottishIsland \sqsubseteq BritishPlace$.

R20 infers individual inequality due to combination of disjoint relations and inverse relations, i.e. $a : \exists r.A$ and $A \sqsubseteq \exists fi(fc(r)).\{b\}$ imply $a \not\doteq b$ because if $a \doteq b$ then there is some $c : A$ s.t. $(a, c) : r$ and $(a, c) : fc(r)$, which is inconsistent.

5.3. TBox Approximate Reasoning Tractability & Soundness

We call the rules **R1-20** (Table 2,4,5 and 6) the **R** rule set. With the **R** rules, TBox approximate reasoning can be performed by repeatedly applying the **R** rules until no new results can be derived. The results of applying these rules is defined as follows. As we can see, the reasoning is initialised with the approximated $\mathcal{EL}^{++}$ TBox and certain tautology axioms.

Definition 5 ***(TBox Approximate Reasoning Closure)*** *For an ontology $\mathcal{O}$ that is transformed into $TA_{fa}(\mathcal{O}) = (\mathcal{T}, CT, QT, IT)$ w.r.t. name assignment $fa(.)$, its corresponding TBox Approximate Reasoning Closure, denoted by $ST(TA_{fa}(\mathcal{O}))$, is the set of axioms computed as follows:*

1. *initialising $ST(TA_{fa}(\mathcal{O}))$ as $\mathcal{T} \cup \{fa(P) \sqsubseteq fa(P) | P$ is a term in $\mathcal{O}\} \cup \{fa(C) \sqsubseteq \top | C$ is a concept term in $\mathcal{O}\}$;*
2. *repeatedly applying the **R** rules on $ST(TA_{fa}(\mathcal{O}))$, CT, QT, IT and adding inferred new axioms into $ST(TA_{fa}(\mathcal{O}))$ until no new results can be inferred;*

Computation of TBox approximate reasoning closure is tractable:

Theorem 4 ***(TBox Approximate Reasoning Complexity)*** *For an ontology $\mathcal{O}$, its TBox Approximate Reasoning Closure $ST(TA_{fa}(\mathcal{O}))$ can be computed in polynomial time w.r.t. the size of $\mathcal{O}$.*

This theorem can be proved by showing that the size of $ST(TA_{fa}(\mathcal{O}))$ is polynomial w.r.t. $|\mathcal{CN}_{\mathcal{T}}| + |\mathcal{RN}_{\mathcal{T}}|$, and it can be computed incrementally.

This theorem shows that the overall complexity of perform TBox syntactic approximate reasoning is PTIME-Complete.

The reasoning is also soundness guaranteed:

Theorem 5 ***(TBox Approximation Reasoning Soundness)*** *For an ontology $\mathcal{O}$, let α be an axiom such that $\Sigma_\alpha \subseteq \Sigma_{\mathcal{O}}$, then $\alpha \in ST(TA_{fa}(\mathcal{O}))$ only if $\mathcal{O} \models \alpha$.*

In Sec. 7 we will also see that, our approach have practically high recall on many benchmark and/or real-world ontologies.

5.4. Nominal-safety Under Approximate Reasoning

As we explained in Sec. 2.2, in this paper we will focus on ontologies whose approximate reasoning closure is nominal-safe. With the notion of TBox approximate reasoning closure, we can now define the notion of nominal-safety under approximate reasoning.

Definition 6 ***(Semantic Nominal-safety Under Approximate Reasoning)*** *An ontology $\mathcal{O}$ is semantic nominal-safe under approximate reasoning if $ST(TA_{fa}(\mathcal{O}))$ is a nominal-safe $\mathcal{EL}^{++}$ ontology (axioms of form $\{a\} \equiv \{b\}$ and $\{a\} \sqcap \{b\} \sqsubseteq \bot$ are allowed).*

Basically, an ontology is safe if its closure is always safe. As we mentioned earlier, the nominal-safety condition is useful to improve the effectiveness of the $\mathcal{EL}^{++}$ completion rules in approximate reasoning. However it requires computation of the closure hence cannot be checked before reasoning. Given the **R** rules, we define a syntactic notion of nominal-safety:

Definition 7 ***(Syntactic Nominal-safety Under Approximate Reasoning)*** *An ontology $\mathcal{O}$ is syntactic nominal-safe under approximate reasoning if $TA_{fa}(\mathcal{O}) = (\mathcal{T}, CT, QT, IT)$ satisfies the following syntactic properties, where A is a concept such that A, $fc(A) \in \mathcal{CN}_{\mathcal{T}}$, $B_{(i)}$ is an arbitrary concept, a and b are individuals and r is a named role:*

1. *$\mathcal{T}$ is a nominal-safe $\mathcal{EL}^{++}$ TBox;*
2. *$fc(\{a\}) \sqsubseteq A \notin \mathcal{T}$;*
3. *$fc(\{a\}) \sqsubseteq \{b\} \notin \mathcal{T}$;*
4. *$B_1 \sqcap \cdots \sqcap fc(\{a\}) \sqcap \cdots \sqcap B_n \sqsubseteq \bot \notin \mathcal{T}$;*
5. *$fc(\{a\}) \sqsubseteq \exists r.A \notin \mathcal{T}$;*
6. *$fc(\{a\}) \sqsubseteq \exists r.\{b\} \notin \mathcal{T}$;*

7. $fc(\{a\}) \sqsubseteq \exists r.fc(\{b\}) \notin \mathcal{T}$;

This syntactic condition can be checked directly on the transformation results. It can be shown that when an ontology is syntactic nominal-safe, it is also semantic nominal-safe:

Theorem 6 ***(Nominal-safety Under Approximate Reasoning)*** *If an ontology $\mathcal{O}$ satisfies the syntactic nominal-safety condition defined in Def. 7, then it is semantic nominal-safe under approximate reasoning, as defined in Def. 6.*

In the rest of the paper, we consider only ontologies that satisfy the syntactic conditions specified in Def. 7 and simply call them *nominal-safe ontologies*. As we will show in the next section, this condition is important to guarantee the quality of some ABox reasoning optimisations.

5.5. TBox Syntactic Approximation Summary

So far, we have presented the definition of TBox Transformation (Def. 4) and the corresponding approximate deduction rules–the **R** rule set. The results of such approximate reasoning is soundness-guaranteed (Theorem 5) and tractable (Theorem 4). In the Evaluation section (Sec. 7) we will conduct experiments on TBox classification and compare with other off-the-shelf reasoners to show that our approach is indeed efficient and yields high recalls.

Comparing to our previous work [65], the transformation and completion rules are extended and revised to support more expressive power such as inverse roles, role disjointness and nominals. We also formally characterised the relation between the orignal ontology and transfomed TBox (Theorem 2). Furthermore, we presented the semantic (Def. 6) and syntactic (Def. 7) conditions of nominal-safety under approximate reasoning. We assume that such conditions are satisfied in the rest of the paper. In the next section, we will extend the transformation and approximate deduction to supporting ABox reasoning.

6. Syntactic Approximation for ABox Reasoning

TBox transformation does not include information about ABox axioms in the original ontology thus can not be used for ABox materialisation. Also, when the TBox contains nominals, ABox materialisation and TBox classification can not always be completely separated. The $\mathcal{T}_4$ (page 19) is an example in which axiom $\{Britain\} \sqsubseteq \exists include.\{Scotland\}$ can be regarded as an ABox axiom.

In this section, we further extend syntactic approximation as an approach to tractable approximate reasoning for ABox. Combined with the TBox approximate reasoning introduced in Sec. 5 we can provide syntactic approximation for the entire ontology. Similar as the last section, we first present the transformation (Sec. 6.1), then the approximate deduction approaches with different optimisations, including internalisation (Sec. 6.2), separate ABox reasoning for nominal-free ontology (Sec. 6.3) and combined TBox and ABox reasoning with nominals (Sec. 6.4).

6.1. ABox Transformation

We perform ABox transformation in a similar manner as the TBox transformation in Def. 4, with an addition of an $\mathcal{EL}^{++}$ ABox:

Definition 8 ***($\mathcal{EL}^{++}_{\mathcal{CQI}}$ Transformation)*** *Given an Ontology $\mathcal{O}$ and a name assignment $fa(\textbf{.})$, let $TA_{fa} = (\mathcal{T}', CT', QT', IT')$ be its TBox transformation, its $\mathcal{EL}^{++}_{\mathcal{CQI}}$* transformation *$A_{fa}(\mathcal{O})$ is a five-tuple $(\mathcal{T}, \mathcal{A}, CT, QT, IT)$ constructed as follows:*

1. $\mathcal{T} = \mathcal{T}'$, $CT = CT'$, $QT = QT'$, $IT = IT'$.
2. *for each* $a : C \in \mathcal{O}$, $\mathcal{A} = \mathcal{A} \cup \{a : fa(C)\}$.
3. *for each* $(a, b) : R \in \mathcal{O}$, $\mathcal{A} = \mathcal{A} \cup \{(a, b) : fa(R)\}$.
4. *for each* $(a, b) : \neg R \in \mathcal{O}$, $\mathcal{A} = \mathcal{A} \cup \{(a, b) : fa(\neg R)\}$
5. *for each axiom* $a \not\doteq b \in \mathcal{O}$, $\mathcal{A} = \mathcal{A} \cup \{a \not\doteq b\}$.
6. *for each axiom* $a \doteq b \in \mathcal{O}$, $\mathcal{A} = \mathcal{A} \cup \{a \doteq b\}$.

Similar as the TBox transformation, the $\mathcal{EL}^{++}_{\mathcal{CQI}}$ transformation will also be used to support the approximate deduction procedures. Step-1 reuses the TBox transformation. Note that in the TBox transformation we have already considered all the terms, even if they appear only in ABox. Thus we do not need to change any of the CT, QT or IT; Step-2 to 6 rewrite all the ABox axioms; Also similar as before, we omit the $\mathcal{EL}^{++}_{\mathcal{CQI}}$ in the notation in the rest of the paper, simply calling it *the transformation* and denoting it with $A_{fa}(\mathcal{O})$

The transformation can be illustrated with the following example:

Example 5 *Ontology $\mathcal{O}_5$ contains TBox $\mathcal{T}_5$ and ABox $\mathcal{A}_5$. $\mathcal{T}_5$ contains axioms:*

$$BritishCity \equiv City \sqcap \exists locatedIn.(EnglishArea \sqcup ScottishArea \sqcup WalesArea) \tag{20}$$

$$locatedIn \circ locatedIn \sqsubseteq locatedIn \tag{21}$$

$\mathcal{A}_5$ contains axioms:

$$aberdeen : City \tag{22}$$

$$(aberdeen, grampian) : locatedIn \tag{23}$$

$$grampian : \exists locatedIn.ScottishArea \tag{24}$$

A name assignment $fa(.)$ assigns the following names (among others):

$$\begin{aligned}
fa(City \sqcap \exists locatedIn.(EnglishArea \sqcup ScottishArea \sqcup WalesArea)) &= C_1 \\
fa(\exists locatedIn.(EnglishArea \sqcup ScottishArea \sqcup WalesArea)) &= C_2 \\
fa(EnglishArea \sqcup ScottishArea \sqcup WalesArea) &= C_3 \\
fa(\neg EnglishArea \sqcup \neg SottishArea \sqcup \neg WalesArea) &= cC_3 \\
fa(\exists locatedIn.ScottishArea) &= C_4 \\
fa(\neg EnglishArea) &= cEArea \\
fa(\neg ScottishArea) &= cSArea \\
fa(\neg WalesArea) &= cWArea \\
fa(\neg\{grampian\}) &= cGrampian \\
fa(\neg\{aberdeen\}) &= cAberdeen
\end{aligned}$$

Then by Def. 8, we have $A_{fa}(\mathcal{O}_5) = (\mathcal{T}_{city}, \mathcal{A}_{city}, CT_{city}, QT_{city}, IT_{city})$, where $\mathcal{T}_{city}$ contains axiom (21) and the following axioms:

$$BritishCity \equiv C_1 \tag{25}$$

$$C_1 \equiv City \sqcap C_2 \tag{26}$$

$$C_2 \equiv \exists locatedIn.C_3 \tag{27}$$

$$cC_3 \equiv cEArea \sqcap cSArea \sqcap cWArea \tag{28}$$

$$C_4 \equiv \exists locatedIn.ScottishArea \tag{29}$$

$\mathcal{A}_{city}$ contains axioms (22), (23) and

$$grampian : C_4 \tag{30}$$

CT_{city} contains corresponding name pairs for complementary terms, such as (C_3, cC_3), $(ScottishArea, cSArea)$. Note that pairs of nominals and their complements such as $(\{aberdeen\}$, $cAberdeen)$ and $(\{grampian\}$, $cGrampian)$ are not included in CT_{city} because $\{aberdeen\}$ and $\{grampian\}$ are not terms of $\mathcal{O}_5$.

QT_{city} is empty.

IT_{city} contains corresponding name pairs for $locatedIn$ and its inverse.

The transformation can also be performed in linear time:

Proposition 7 ***(Linear Transformation)*** *In Def. 8, the transformation from $\mathcal{O}$ to $A_{fa}(\mathcal{O})$ can be done in linear time.*

Similar as the TBox transformation, this transformation results can also be regarded as a syntactic variants of the original ontology that preserves reasoning of the original signature:

Theorem 8 ***(Ontology Reasoning Preservation)*** *For any ontology $\mathcal{O}$ and its transformation $(\mathcal{T}, \mathcal{A}, CT, QT, IT)$, let $\mathcal{O}' = (\mathcal{T}', \mathcal{A}')$ be an ontology constructed as follows:*

1. *$\mathcal{T}'$ is constructed in the same way as in Theorem 2;*
2. *$\mathcal{A}' = \mathcal{A}$;*

then for any axiom α with $\Sigma_\alpha \subseteq \Sigma_\mathcal{O}$, we have $\mathcal{O} \models \alpha$ iff $\mathcal{O}' \models \alpha$:

Similar as Theorem 2, this theorem can be proved by showing that $\mathcal{O} \cup \mathcal{O}' \cup \mathcal{T}_{fa}$ conservatively extends both $\mathcal{O}$ and $\mathcal{O}'$ on $\Sigma_\mathcal{O}$, where $\mathcal{T}_{fa}$ is the set of definition $fa(P) \equiv P$ for all term P of $\mathcal{O}$ except complement of roles. The full proof is deferred to the appendix. This theorem also indicates that the transformation itself fully preserves the semantics of the original ontology, while the $(\mathcal{T}, \mathcal{A})$ is a naive soundness preserving $\mathcal{EL}^{++}$ approximation of the original ontology:

Proposition 9 ***(Approximation)*** *For an Ontology $\mathcal{O}$, let its transformation results $A_{fa}(\mathcal{O})$ be $(\mathcal{T}, \mathcal{A}, CT, QT, IT)$ as specified in Def. 8, then $(\mathcal{T}, \mathcal{A})$ is an $\mathcal{EL}^{++}$ ontology such that for any axiom α with $\Sigma_\alpha \subseteq \Sigma_\mathcal{O}$, we have $(\mathcal{T}, \mathcal{A}) \models \alpha$ only if $\mathcal{O} \models \alpha$.*

6.2. ABox Approximate Deduction via Internalisation

ABox reasoning can be done as in classical $\mathcal{EL}^{++}$ (cf. Sec. 3) by encoding the ABox as a concept. However this approach introduces a rather complex concept and will introduce additional concept names in the normalisation phase, making reasoning more difficult. Also, for each instance checking, a separate subsumption checking needs to be performed, making it less efficient for materialisation. Alternatively, we can do the following internalisation:

Definition 9 ***(ABox Internalisation)*** *Given an ontology $\mathcal{O}$ and its transformation $A_{fa}(\mathcal{O}) = (\mathcal{T}', \mathcal{A}', CT', QT', IT')$, its* ABox internalisation*, denoted by $AI(A_{fa}(\mathcal{O}))$, is a four-tuple $(\mathcal{T}, CT, QT, IT)$ constructed as follows:*

1. *$\mathcal{T}$ is initialised as $\mathcal{T}'$, CT is initialised as CT'.*
2. *$QT = QT'$, $IT = IT'$.*

3. *for each* $a \in \mathcal{IN}_{\mathcal{O}}$*,* $CT = \{(\{a\}, fa(\neg\{a\})), (fa(\neg\{a\}), \{a\})\}$*.*
4. *for each* $a \in \mathcal{IN}_{\mathcal{O}}$*,* $\mathcal{T} = \mathcal{T} \cup \{\{a\} \sqsubseteq \{a\}, fc(\{a\}) \sqsubseteq fc(\{a\}), \{a\} \sqsubseteq \top, fc(\{a\}) \sqsubseteq \top\}$*.*
5. *for each* $a : A \in \mathcal{A}'$*,* $\mathcal{T} = \mathcal{T} \cup \{\{a\} \sqsubseteq A\}$*.*
6. *for each* $(a, b) : r \in \mathcal{A}'$*,* $\mathcal{T} = \mathcal{T} \cup \{\{a\} \sqsubseteq \exists r.\{b\}\}$*.*
7. *for each* $a \doteq b \in \mathcal{A}'$*,* $\mathcal{T} = \mathcal{T} \cup \{\{a\} \equiv \{b\}\}$*.*
8. *for each* $a \not\doteq b \in \mathcal{A}'$*,* $\mathcal{T} = \mathcal{T} \cup \{\{a\} \sqsubseteq fc(\{b\})\}$*.*

In this definition, all ABox axioms are internalised into the approximated $\mathcal{EL}^{++}$ TBox. And singleton of each individual and its complement are now treated as terms of the ontology.

Applying the above definition on the transformation described in Example 5 yields the following results:

Example 6 *$AI(A_{fa}(\mathcal{O}_5)) = (\mathcal{T}'_{city}, CT'_{city}, QT_{city}, IT_{city})$, where $\mathcal{T}'_{city}$ includes $\mathcal{T}_{city}$ and the following axioms (among the others):*

$$\{aberdeen\} \sqsubseteq City \tag{31}$$

$$\{aberdeen\} \sqsubseteq \exists locatedIn.\{grampian\} \tag{32}$$

$$\{grampian\} \sqsubseteq C_4 \tag{33}$$

$CT'_{city} = CT_{city} \cup \{(\{aberdeen\}, cAberdeen), (cAberdeen, \{aberdeen\}), (\{grampian\}, cGrampian), (cGrampian, \{grampian\})\}$.

It's easy to show that such internalisation can be constructed in linear time and the result $(\mathcal{T}, CT, QT, IT)$ is still linear w.r.t. the size of $\mathcal{O}$. Also, $\mathcal{T}$ is normalised if $\mathcal{T}'$ normalised. So $AI(A_{fa}(\mathcal{O}))$ is syntactically similar to a TBox transformation. To this end, we reduce ABox reasoning to TBox reasoning on $\mathcal{T}$ and the same **R** rule set can be used. The result of applying these rules is defined as follows:

Definition 10 ***(Approximate Reasoning Closure via Internalisation)*** *For an ontology $\mathcal{O}$, let $AI(A_{fa}(\mathcal{O}))$ be its ABox internalisation, its corresponding Approximate Reasoning Closure via Internalisation, denoted by $SI_{fa}(\mathcal{O})$, is computed as $SI_{fa}(\mathcal{O}) = ST(AI(A_{fa}(\mathcal{O})))$.*

Note that although many nominals internalised from individuals are not terms of the original $\mathcal{O}$, hence will not be used to initialise some tautology axioms as in step-1 of Def. 5, such axioms will still be included in the initialisation of the closure due to step-4 of Def. 9.

The computation of the closure is tractable and soundness-preserving:

Theorem 10 ***(Complexity)*** *Given an ontology $\mathcal{O}$, its approximate reasoning closure via internalisation $SI_{fa}(\mathcal{O})$ can be computed in polynomial time w.r.t. the size of $\mathcal{O}$.*

This theorem can be proved by showing that the internalisation described in Def. 9 is tractable in terms of both time and size.

Theorem 11 ***(Approximate Reasoning via Internalisation Soundness)*** *For an ontology $\mathcal{O}$, let α be an axiom with $\Sigma_\alpha \subseteq \Sigma_{\mathcal{O}}$, then $\alpha \in SI_{fa}(\mathcal{O})$ only if $\mathcal{O} \models \alpha$.*

Note that the closure is a nominal-safe $\mathcal{EL}^{++}$ ontology if the original ontology $\mathcal{O}$ is nominal-safe (Theorem 6), because the internalised ABox axioms will not affect the nominal-safety of the TBox. Also note that the closure contains not only concept subsumptions, but also concept assertions in form of $\{a\} \sqsubseteq A$ and role assertions in form of $\{a\} \sqsubseteq \exists r.\{b\}$. For example, we can infer that $\{aberdeen\} \sqsubseteq BritishCity \in SI_{fa}(\mathcal{O}_5)$. First applying **R3** on axiom (29) and (33) derives $\{grampian\} \sqsubseteq \exists locatedIn.ScottishArea$. Applying **R8** on this axiom and axiom (21) and (32) derives

$$\{aberdeen\} \sqsubseteq \exists locatedIn.ScottishArea. \tag{34}$$

Applying **R10** on axiom (28) we have $ScottishArea \sqsubseteq C_3$. Combining with axiom (27) and the above axiom (34) we can apply **R4** to derive $\{aberdeen\} \sqsubseteq C_2$. Further combining with axiom (26) and (31) we can apply **R2** to derive $\{aberdeen\} \sqsubseteq C_1$. Eventually, combining with axiom (25) we can apply **R1** to derive $\{aberdeen\} \sqsubseteq BritishCity$.

The ABox internalisation internalises the entire ABox into the TBox and reduces ABox reasoning to TBox reasoning. However, a critical limitation is that, treating individuals as nominals will significantly increase the number of terms in the transformation result (Step-3 and 4 of Def. 9) and produce many intermediate results that are neither desired by nor contributing to ontology materialisation. For example, when $\{a\} \sqsubseteq A$ is entailed, another entailment $fc(A) \sqsubseteq fc(\{a\})$ will be derived by rule **R10**. As we will show later, such entailments do not contribute more than what $\{a\} \sqsubseteq A$ can offer. With a large amount of individuals , such non-desired intermediate results require a lot of computation and should be minimised.

6.3. ABox Approximate Deduction for Nominal-free Ontologies

To optimise the performance we devise more fine-grained reasoning mechanisms for combined TBox and ABox. In this sub-section, we start from a simpler case, in which the transformed TBox contains no nominals, such as the $\mathcal{O}_5$ in Example 5. In this case, the ABox reasoning has no effect on the TBox reasoning, which can thus be precomputed. After that, ABox materialisation can be performed directly using ABox axioms and TBox materialisation closure without internalisation. To do that, the nominal-free ABox completion rules in Table 7 are required. In rule **AR1d**, **AR1e** and **AR10**, we have $\sharp \in \{\doteq, \neq\}$.

We call the rules **AR1-19** (in Table 7) the **AR** rules. Each rule **ARi(a,b,c,d,e)** is analogous to the corresponding TBox completion rule **Ri**.

In reasoning, the execution of rules are performed as in the following definition:

Definition 11 ***(Nominal-free Approximate Reasoning Closure)*** *For an ontology $\mathcal{O}$ that is transformed into $A_{fa}(\mathcal{O}) = (\mathcal{T}, \mathcal{A}, CT, QT, IT)$ w.r.t. name assignment $fa(.)$, its corresponding nominal-free approximate reasoning closure, denoted by $SNF_{fa}(\mathcal{O})$, is the set of axioms computed as follows:*

1. *initialising $SNF_{fa}(\mathcal{O})$ as $ST((\mathcal{T}, CT, QT, IT)) \cup \mathcal{A} \cup \{a \doteq a, a : \top | a \in \mathcal{IN}_{\mathcal{O}}\}$;*
2. *repeatedly applying the **AR** rules on $SNF_{fa}(\mathcal{O})$, CT, QT, IT and adding inferred new axioms into $SNF_{fa}(\mathcal{O})$ until no new results can be inferred;*

Step-1 in the above definition shows that TBox materialisation using the **R** rules should be performed before computing the ABox materialisation. Note that duo to the absence of nominals, rules **R6**, **12**, **18**, **19** and **20** won't be executed.

The nominal-free closure also contains desired materialisation results. For example, with the same ontology $\mathcal{O}_5$ and its transformation $A_{fa}(\mathcal{O}_5)$ as in Example 5, we can derive $aberdeen$:

Table 7: ABox completion rules (no datatypes)

AR1a	If $x : A$ and $A \sqsubseteq B$ then $x : B$
AR1b	If $x : A$ and $x \doteq y$ then $y : A$
AR1c	If $x : A$ and $y : fc(A)$ then $x \not\doteq y$
AR1d	If $x \doteq y$ and $y \sharp z$ then $x \sharp z$
AR1e	If $x : \bot$ and y is an individual, then $x \doteq y$ and $x \not\doteq y$
AR2	If $x : A_1, \ldots, x : A_n$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq B$ then $x : B$
AR3a	If $x : A$ and $A \sqsubseteq \exists r.B$ then $x : \exists r.B$
AR3b	If $x \doteq y$ and $y : \exists r.A$ then $x : \exists r.A$
AR3c	If $(x, y) : r$ and $x \doteq z$ then $(z, y) : r$
AR4a	If $x : \exists r.A$, $A \sqsubseteq A'$ and $\exists r.A' \sqsubseteq B$ then $x : B$
AR4b	If $(x, y) : r$, $y : A$ and $\exists r.A \sqsubseteq B$ then $x : B$
AR5a	If $x : \exists r.A$ and $A \sqsubseteq \bot$ then $x : \bot$
AR5b	If $(x, y) : r$ and $y : \bot$ then $x : \bot$
AR7a	If $x : \exists r.A$ and $r \sqsubseteq s$ then $x : \exists s.A$
AR7b	If $(x, y) : r$ and $r \sqsubseteq s$ then $(x, y) : s$
AR8a	If $x : \exists r_1.A$, $A \sqsubseteq \exists r_2.B$ and $r_1 \circ r_2 \sqsubseteq r_3$, then $x : \exists r_3.B$
AR8b	If $(x, y) : r_1$, $y : \exists r_2.B$ and $r_1 \circ r_2 \sqsubseteq r_3$, then $x : \exists r_3.B$
AR8c	If $(x, y) : r_1$, $(y, z) : r_2$ and $r_1 \circ r_2 \sqsubseteq r_3$, then $(x, z) : r_3$
AR9a	If $x : A, B$ and $A = fc(B)$ then $x : \bot$
AR9b	If $x \doteq y$ and $x \not\doteq y$ then $x : \bot$
AR10	If $x \sharp y$, then $y \sharp x$
AR11	If $x : A_1, \ldots, x : A_{i-1}, x : A_{i+1}, \ldots, x : A_n$ and $\bigsqcap_{1 \leq j \leq n} A_j \sqsubseteq \bot$ then $x : fc(A_i)$
AR12	If $(x, y) : r$ and $(z, y) : fc(r)$ then $x \not\doteq z$
AR19	If $(x, y) : r$ then $(y, x) : fi(r)$

$BritishCity \in SNF_{fa}(\mathcal{O}_5)$. First applying **AR3a** on axiom (29) and (30) derives $grampian : \exists locatedIn.ScottishArea$. Applying **AR8b** on this axiom and axiom (21) and (23) derives

$$aberdeen : \exists locatedIn.ScottishArea. \tag{35}$$

Applying **R10** on axiom (28) we have $ScottishArea \sqsubseteq C_3$. Combining with axiom (27) and the above axiom (35) we can apply **AR4a** to derive $aberdeen : C_2$. Further combining with axiom (22) and (26) we can apply **AR2** to derive $aberdeen : C_1$. Eventually, combining with axiom (25) we can apply **AR1a** to derive $aberdeen : BritishCity$. The procedure and result are similar to deriving $\{aberdeen\} \sqsubseteq BritishCity \in SI_{fa}(\mathcal{O}_5)$.

Compared with the internalisation approach, nominals are not used. Consequently, entailments involving *Complements of Nominals* (CoNs for short, e.g. $fc(\{a\})$) are not computed at all. This reduces the size of the closure and improves efficiency.

The analogousness of completion rules indicates that similar algorithms can be applied and tractability and soundness are preserved. We show that this approach should be as complete as the internalisation approach when the ontology contains no nominal:

Theorem 12 *For a nominal-free ontology $\mathcal{O}$, let $a, b \in \mathcal{IN}_{\mathcal{O}}$ be two individuals, $A, B \in \mathcal{CN}_{\mathcal{O}}$ be two concepts, and $r \in \mathcal{RN}_{\mathcal{O}}$ be a role, then the following holds:*

1. $A \sqsubseteq B \in SI_{fa}(\mathcal{O})$ iff $A \sqsubseteq B \in SNF_{fa}(\mathcal{O})$*;*
2. $\{a\} \sqsubseteq A \in SI_{fa}(\mathcal{O})$ iff $a : A \in SNF_{fa}(\mathcal{O})$*;*
3. $\{a\} \sqsubseteq \exists r.\{b\} \in SI_{fa}(\mathcal{O})$ iff $(a, b) : r \in SNF_{fa}(\mathcal{O})$*;*

The ← direction is obvious because the **AR** rules are special cases of the corresponding **R** rules where certain atomic concepts are restricted to nominals.

The → directions of the theorem can be shown by proving the following stronger lemma:

Lemma 13 *For a nominal-free ontology $\mathcal{O}$, let $A_{fa}(\mathcal{O}) = (\mathcal{T}, \mathcal{A}, CT, QT, IT)$ and $AI(A_{fa}(\mathcal{O}))$ be $(\mathcal{T}', CT, QT, IT)$. If we use SI to denote $SI_{fa}(\mathcal{O})$, ST to denote $ST((\mathcal{T}, CT, QT, IT))$, and SNF to denote $SNF_{fa}(\mathcal{O})$, then the following invariants hold in each step of computation of SI, where $a, b \in \mathcal{IN}_{\mathcal{O}}$ are two individuals, $A, B \in \mathcal{CN}_{\mathcal{T}} \cup \{\top, \bot\}$ are two concepts, and $r \in \mathcal{RN}_{\mathcal{O}}$ is a role:*

$$A \sqsubseteq B \in SI \textit{ only if } A \sqsubseteq B \in ST \tag{36}$$

$$A \sqsubseteq fc(\{a\}) \in SI \textit{ only if } a : fc(A) \in SNF \tag{37}$$

$$\{a\} \sqsubseteq B \in SI \textit{ only if } a : B \in SNF \tag{38}$$

$$\{a\} \sqsubseteq \{b\} \in SI \textit{ only if } a \doteq b \in SNF \tag{39}$$

$$\{a\} \sqsubseteq fc(\{b\}) \in SI \textit{ only if } a \not\doteq b \in SNF \tag{40}$$

$$fc(\{a\}) \sqsubseteq fc(\{b\}) \in SI \textit{ only if } a \doteq b \in SNF \tag{41}$$

$$A \sqsubseteq \bot \notin SI \textit{ and } A \sqsubseteq \exists r.B \in SI \textit{ only if } A \sqsubseteq \exists r.B \in ST \tag{42}$$

$$\{a\} \sqsubseteq \bot \notin SI \textit{ and } \{a\} \sqsubseteq \exists r.B \in SI \textit{ only if } a : \exists r.B \in SNF \tag{43}$$

$$\{a\} \sqsubseteq \exists r.\{b\} \in SI \textit{ only if } (a, b) : r \in SNF \tag{44}$$

Apparently the → directions of the theorem hold if the above lemma holds. This lemma can further be proved by induction on application of rules. A detailed proof is deferred to the Appendix.

Theorem 12 shows that when the original ontology is nominal-free, the combination of the **R** rules and **AR** rules without using CoNs in transformation and reasoning can achieve as complete results as the internalisation approach. When the materialisation contains a lot of class assertion axioms of individuals, this significantly reduces the memory and computation cost of reasoning.

The Lemma 13 also shows that internalising an individual a into nominal $\{a\}$ alone will not introduce unnecessary inference results, because for each inferred axiom involving $\{a\}$ in internalisation there is a corresponding inferred axiom involving a without internalisation. The major source of unnecessary results is the introduction of CoNs. Particularly, invariant 37 and invariant 41 suggest that certain axioms involving CoNs are redundant as they can be replaced by other inferred axioms. In the next section, we will exploit this observation to develop more fine-grained algorithms for reasoning with nominals.

6.3.1. Optimising for Ontologies with Transitive-only Role Chains

We can further restrict the syntax of ontology to simplify the intermediate results of materialisation. Particularly, if the original ontology contains only one type of role chain – the transitive role, then we can avoid computing axioms of form $a : \exists r.A$ at all in ABox materialisation. More precisely, in the approximate deduction of such an ontology, rules **AR3a, 3b, 4a, 5a, 7a, 8a, 8b** can be replaced by the alternative rules illustrated in Table 8.

Table 8: Alternative ABox completion rules for transitive-only role chains

TR4a	If $(x, y) : r$, $y : A$ and $\exists r.A \sqsubseteq B$ then $x : B$
TR4b	If $(x, y) : r$, $y : A$, $A \sqsubseteq \exists r.A'$, $r \circ r \sqsubseteq r$, $r \sqsubseteq s$, $A' \sqsubseteq A''$ and $\exists s.A'' \sqsubseteq B$ then $x : B$

We use the **TR** rules to denote the set containing **AR1(a,b,c,d,e)**, **AR2**, **AR3c**, **AR4b**, **TR4(a,b)**, **AR5b**, **AR7b**, **AR8c**, **AR9(a,b)**, **AR10**, **AR11**, **AR12**, **AR19**. In other words, the **TR** rule set is the **AR** rule set after replacing rules involving axioms of form $a : \exists r.A$ with the two alternative rules in Table 8.

In reasoning, the execution of rules are performed as in the following definition:

Definition 12 ***(Transitive-only Nominal-free Approximate Reasoning Closure)*** *For a nominal-free ontology $\mathcal{O}$ that is transformed into $A_{fa}(\mathcal{O}) = (\mathcal{T}, \mathcal{A}, CT, QT, IT)$ w.r.t. name assignment $fa(\textbf{.})$, its corresponding transitive-only nominal-free approximate reasoning closure, denoted by $STNF_{fa}(\mathcal{O})$, is the set of axioms computed as follows:*

1. *initialising $STNF_{fa}(\mathcal{O})$ as $ST((\mathcal{T}, CT, QT, IT)) \cup \mathcal{A} \cup \{a \doteq a, a : \top | a \in \mathcal{IN}_{\mathcal{O}}\}$;*
2. *repeatedly applying the **TR** rules on $STNF_{fa}(\mathcal{O})$, CT, QT, IT and adding inferred new axioms into $STNF_{fa}(\mathcal{O})$ until no new results can be inferred;*

As can be seen from the definition, the transitive-only closure (Def. 12) is similar as the normal nominal-free closure (Def. 11) except that the former is closed under the **TR** rules while the later is closed under the **AR** rules.

The **TR** rules can also compute desired materialisation results. For example, the $\mathcal{O}_5$ in Example 5 is an ontology with transitive-only role chains. We can derive $aberdeen : BritishCity \in STNF_{fa}(\mathcal{O}_5)$ with the **TR** rules. First we can derive $ScottishArea \sqsubseteq C_3$ in the same way as before. Then we apply **TR4b** on this axiom and axioms (21), (23), (27), (29) and (30), which yield $aberdeen : C_2$. After that we can derive $aberdeen : BritishCity$ in the same way as before.

Compared with the **AR** rules, entailments involving $a : \exists r.A$ are not computed at all. This reduces the size of the closure and improves efficiency.

The tractability and soundness of computing the transitive-only nominal-free closure is quite obvious. We show that this approach should be as complete as the normal nominal-free approximate reasoning approach when the ontology contains no other role chains except the role transitivity:

Theorem 14 *For a nominal-free ontology $\mathcal{O}$, let $a, b \in \mathcal{IN}_{\mathcal{O}}$ be two individuals, $A, B \in \mathcal{CN}_{\mathcal{O}}$ be two concepts, and $r \in \mathcal{RN}_{\mathcal{O}}$ be a role, if for any $r_1 \circ \ldots r_n \sqsubseteq s \in \mathcal{O}$, we have $n = 2$ and $r_1 = \cdots = r_n = s$, then the following holds:*

1. $A \sqsubseteq B \in SNF_{fa}(\mathcal{O})$ iff $A \sqsubseteq B \in STNF_{fa}(\mathcal{O})$*;*
2. $\{a\} \sqsubseteq A \in SNF_{fa}(\mathcal{O})$ iff $a : A \in STNF_{fa}(\mathcal{O})$*;*
3. $\{a\} \sqsubseteq \exists r.\{b\} \in SNF_{fa}(\mathcal{O})$ iff $(a, b) : r \in STNF_{fa}(\mathcal{O})$*;*

This theorem can be proved in a similar manner as Theorem 12, by showing some invariants with induction on rule applications.

The above theorem shows that when a nominal-free ontology contains no role chains except the role transitivity axioms, its ABox approximate reasoning can be significantly simplified due to the fact that intermediate results of form $a : \exists r.A$ no longer need to be computed and maintained. Such an optimisation will not affect the quality of reasoning and is rather practical as many of the existing real-world ontologies do not employ complex role chains.

6.4. Combined TBox and ABox Approximate Deduction with Nominals

ABox completion presented in Sec. 6.3 has a restriction that the approximated TBox should contain no nominals. Following our objective of combining TBox and ABox approximate reasoning, in this section we further investigate how to handle ABox when TBox contains nominals. Considering the following ontology:

Example 7 *Ontology $\mathcal{O}_7$ contains TBox $\mathcal{T}_7$ and ABox $\mathcal{A}_7$. $\mathcal{T}_7$ contains axiom (21) and two other axioms as follows:*

$$BritishCity \equiv City \sqcap \exists locatedIn.\{england, scotland, wales\} \tag{45}$$

$$ScottishArea \sqsubseteq \exists locatedIn.\{scotland\} \tag{46}$$

$$locatedIn \circ locatedIn \sqsubseteq locatedIn$$

$\mathcal{A}_7$ contains axiom (22) and another axiom as follows:

$$aberdeen : City$$

$$aberdeen : \exists locatedIn.ScottishArea \tag{47}$$

A name assignment $fa(\boldsymbol{.})$ assigns the following names (among others):

$$\begin{aligned}
fa(City \sqcap \exists locatedIn.\{england, scotland, wales\}) &= C_5\\
fa(\exists locatedIn.\{england, scotland, wales\}) &= C_6\\
fa(\{england, scotland, wales\}) &= C_7\\
fa(\neg\{england\} \sqcap \neg\{scotland\} \sqcap \neg\{wales\}) &= cC_7\\
fa(\exists locatedIn.\{scotland\}) &= C_8\\
fa(\exists locatedIn.ScottishArea) &= C_9\\
fa(\neg\{england\}) &= cEngland\\
fa(\neg\{scotland\}) &= cScotland\\
fa(\neg\{wales\}) &= cWales\\
fa(\neg\{grampian\}) &= cGrampian\\
fa(\neg\{aberdeen\}) &= cAberdeen
\end{aligned}$$

Then by Def. 8, we have $A_{fa}(\mathcal{O}_7) = (\mathcal{T}^*_{city}, \mathcal{A}^*_{city}, CT^*_{city}, QT^*_{city}, IT^*_{city})$, where $\mathcal{T}^*_{city}$ con-

tains axiom (21) and the following axioms:

$$BritishCity \equiv C_5 \tag{48}$$
$$C_5 \equiv City \sqcap C_6 \tag{49}$$
$$C_6 \equiv \exists locatedIn.C_7 \tag{50}$$
$$cC_7 \equiv cEngland \sqcap cScotland \sqcap cWales \tag{51}$$
$$ScottishArea \sqsubseteq C_8 \tag{52}$$
$$C_8 \equiv \exists locatedIn.\{scotland\} \tag{53}$$
$$C_9 \equiv \exists locatedIn.ScottishArea \tag{54}$$

$\mathcal{A}^*_{city}$ *contains axiom (22) and the following axiom:*

$$aberdeen : C_9 \tag{55}$$

CT^*_{city} *contains name pairs for complementary terms, such as* $(\{england\}, cEngland)$, $(\{scotland\}, cScotland)$, $(\{wales\}, cWales)$, (C_7, cC_7). *Note that* $(\{aberdeen\}, cAberdeen)$ *is not included in* CT_{city} *because* $\{aberdeen\}$ *is not a term of* $\mathcal{O}_7$.

QT^*_{city} *is empty.*

IT^*_{city} *contains corresponding name pairs for* $locatedIn$ *and its inverse.*

This ontology contains nominals in the TBox. Using the internalisation approach will introduce a new nominal $\{aberdeen\}$ and new CoN $cAberdeen$, which will trigger further unnecessary inference, e.g. deriving $fc(BritishCity) \sqsubseteq cAberdeen$.

One naive strategy for materialising such ontologies is to partition the transformed ABox $\mathcal{A}$ into two disjoint-union subsets $\mathcal{A}_I$ and $\mathcal{A}_E$ such that $\mathcal{A}_I$ contains all axioms that are syntactically reachable from the nominals [5] while $\mathcal{A}_E$ contains the others. When doing reasoning, $\mathcal{A}_I$ can be internalised into the TBox to compute a closure under the **R** rules. Then the closure can be combined with the $\mathcal{A}_E$ to further compute a closure under the **AR** rules. Because $\mathcal{A}_I$ contains all nominal reachable ABox axioms and $\mathcal{A}_E$ contains none, it is easy to realise that such a closure should contain the same ontology materialisation results as the internalisation approach presented in Sec. 6.2. But the individuals in $\mathcal{A}_E$ do not need to be treated as nominals and the reasoning can be simplified for these individuals as the approach presented in Sec. 6.3.

This partitioning-internalisation approach is similar to the spirit of the modularisation-based classification recently presented by Armas Romero et. al. [67] and provides a means to combine the approximate reasoning of TBox and ABox when nominals present. However, it still has its limitations when applied on real-world ontologies. Particularly, when the ontology is tightly coupled, the nominal-reachable partitioning becomes too coarser, resulting in $\mathcal{A}_I$ almost as large as $\mathcal{A}$. In this case, the reasoner has to internalise almost the entire ABox, making the partitioning almost useless. For example, the nominal reachable part of $\mathcal{A}^*_{city}$ actually contains all its axioms.

Recall that Lemma 13 suggests that the introduction of CoNs, i.e. concepts of form $\neg\{a\}$, is the major source of redundancy in the internalisation approach. Hence they should be minimised. In what follows, we show that when the $\mathcal{EL}^{++}$ ontology obtained through transformation satisfies the nominal-safe criteria described in Theorem 6, TBox and ABox reasoning can be combined

[5] An axiom is syntactically nominal-reachable if it contains a nominal, or it contains a concept or role name that appears in another syntcatically nominal-reachable axiom.

and no non-native CoN is needed in reasoning, where a CoN $\neg\{a\}$ is *native* w.r.t. an ontology $\mathcal{O}$ *iff* it is a term of $\mathcal{O}$. Note that in the ABox internalisation approach (Def. 9) many non-native CoNs can be created in addition to the native terms.

Such a combined reasoning approach involves the following major components:

1. In order to classify TBox with nominals, the **R** rules will be needed. Especially, in **R18**, the condition $\{a_i\} \sqsubseteq \exists s.\{b_j\}$ $(1 \leq i \leq n, 1 \leq j \leq m)$ should be changed to (a_i, b_j) : s $(1 \leq i \leq n, 1 \leq j \leq m)$ because now these relations are all converted into ABox.
2. Similarly, the **AR** rules will all be needed.
3. In order to reuse the above rules as much as possible, rules to convert axioms in forms with and without nominals are needed (Table 9). Each of these rules can be executed on both the $\rightarrow$ direction and the $\leftarrow$ direction, respectively. The $\leftarrow$ direction can be regarded as internalisation while the $\rightarrow$ direction can be regarded as reverse internalisation.
   In order to minimise the use of nominals, each of these rules can be applied only if corresponding nominals have already been introduced in inferences. For example, when we have derived $A \sqsubseteq \exists r.\{a\}$, $a \doteq b$ and $\exists r.\{b\} \sqsubseteq B$, it is safe to apply **IR3** to rewrite $a \doteq b$ into $\{a\} \sqsubseteq \{b\}$ because both $\{a\}$ and $\{b\}$ have already been introduced in other axioms. This helps to reuse **R4** to derive $A \sqsubseteq B$ without introducing additional nominals. Note that native nominals are always introduced.
   When introduced nominals are much fewer than individuals, such duplication of entailments will not significantly increase the size of reasoning closure. Nevertheless it's worth mentioning that in reasoner implementation it is easy to use one of the two forms to represent the other to reduce redundancy. Consequently, syntactic variations of corresponding rules need to be implemented as well.
4. The above **R** rules and **AR** rules do not deal with nominals in axioms. To handle them, additional rules illustrated in Table 10 are needed. Note that **CR8a** introduces non-native nominals when certain semantic criteria are met. Nevertheless, their introduction is not accompanied by introduction of any CoN, thus shall not introduce redundancy.

Table 9: nominal-related axiom conversion rules

IR1	$\{x\} \sqsubseteq A$	$\leftrightarrow$	$x : A$
IR2	$\{x\} \sqsubseteq \exists r.\{y\}$	$\leftrightarrow$	$(x, y) : r$
IR3	$\{x\} \sqsubseteq \{y\}$	$\leftrightarrow$	$x \doteq y$
IR4	$\{x\} \sqsubseteq fc(\{y\})$	$\leftrightarrow$	$x \not\doteq y$
IR5	$\{x\} \sqsubseteq \exists r.A$	$\leftrightarrow$	$x : \exists r.A$
IR6	$x : \{y\}$	$\leftrightarrow$	$x \doteq y$
IR7	$x : fc(\{y\})$	$\leftrightarrow$	$x \not\doteq y$
IR8	$x : \exists r.\{y\}$	$\leftrightarrow$	$(x, y) : r$

We use the name **CR** rules to denote the set of rules containing all the **R** rules (Table 2,4,5 and 6), **AR** rules (Table 7), **IR1-8** rules (Table 9) and the rules in Table 10. In reasoning the executions of the **R** rules yield the following closure:

Definition 13 ***(Combined Approximate Reasoning Closure)*** *For a ontology $\mathcal{O}$ that is transformed to $A_{fa}(\mathcal{O}) = (\mathcal{T}, \mathcal{A}, CT, QT, IT)$ w.r.t. a name assignment $fa(\textbf{.})$, its corresponding Combined Approximate Reasoning Closure, denoted by $SC_{fa}(\mathcal{O})$, is the set of axioms computed as follows:*

Table 10: Combined completion rules

CR2	If $\bigsqcap_{1 \leq i \leq n} A_i \sqsubseteq fc(\{b\})$, for all $1 \leq i \leq n$, either $\top \sqsubseteq A_i$, or $A_i = fc(\{a_i\})$ and $a \doteq a_i$, then $a \doteq b$
CR8a	If $A \sqsubseteq \exists r_1.\{a\}$, $(a, b) : r_2$ and $r_1 \circ r_2 \sqsubseteq r_3$ then $A \sqsubseteq \exists r_3.\{b\}$
CR8b	If $a : \exists r_1.A$, $A \sqsubseteq \exists r_2.\{b\}$ and $r_1 \circ r_2 \sqsubseteq r_3$ then $(a, b) : r$
CR11	If $\bigsqcap_{1 \leq i \leq n} A_i \sqsubseteq \bot$, for all $j = 1, \ldots, i-1, i+1, \ldots, n$, either $\top \sqsubseteq A_j$ or $A_j = fc(\{a_j\})$ and $a \doteq a_j$, then $a : A_i$
CR12	If $A \sqsubseteq \exists r.\{b\}$ and $(a, b) : fc(r)$, then $a : fc(A)$
CR20	If $a : \exists r.B$ and $B \sqsubseteq \exists fi(fc(r)).\{b\}$, then $a \neq b$

1. *initialising $SC_{fa}(\mathcal{O})$ as $\mathcal{T} \cup \mathcal{A}$;*
2. *for each term P of $\mathcal{O}$, $SC_{fa}(\mathcal{O}) = SC_{fa}(\mathcal{O}) \cup \{fa(P) \subseteq fa(P)\}$;*
3. *for each concept term C of $\mathcal{O}$, $SC_{fa}(\mathcal{O}) = SC_{fa}(\mathcal{O}) \cup \{fa(C) \sqsubseteq \top\}$;*
4. *for each individual a in $\mathcal{O}$, $SC_{fa}(\mathcal{O}) = SC_{fa}(\mathcal{O}) \cup \{a \doteq a, a : \top\}$;*
5. *repeatedly applying the* ***CR*** *rules on $SC_{fa}(\mathcal{O})$, CT, QT, IT and adding inferred new axioms into $SC_{fa}(\mathcal{O})$ until no new results can be inferred;*

As can be seen from the above definition, the combined closure (Def. 13) is similar to the combination of the TBox closure (Def. 5) and nominal-free ABox closure (Def. 11) except that this time TBox and ABox reasonings are not separated, and more importantly, non-native CoNs are not introduced as in step-3 and 4 of Def. 9.

The **CR** rules produce the desired results in the closure. For example, $aberdeen : BritishCity \in SC_{fa}(\mathcal{O}_7)$ can be derived without introducing non-native CoN $\neg\{aberdeen\}$. First of all, we can re-derive axiom (47) by applying **AR3a** on axiom (54) and axiom (55). Then we can re-derive axiom (46) by applying **R3** on axiom (52) and (53). Combining these two axioms with axiom (21), applying **CR8b** we can infer

$$(aberdeen, scotland) : locatedIn \tag{56}$$

On the other hand, applying **R10** on axiom (51) yields that $\{scotland\} \sqsubseteq C_7$. Combining this with axiom (50) and the above axiom (56), applying **AR4b** we can derive $aberdeen : C_6$. Together with axiom (22) we can further derive $aberdeen : C_5$. Together with axiom (48) we can eventually derive $aberdeen : BritishCity$. As we can see, in this inference procedure, $\{aberdeen\}$ or $\neg\{aberdeen\}$ does not need to be introduced.

Given the forms of the additional rules in the **CR** rule set, it is apparent that soundness and tractability of reasoning can be guaranteed. Furthermore, the results should be as complete as the internalisation approach:

Theorem 15 *For a nominal-safe ontology $\mathcal{O}$ under approximate reasoning, let $a, b \in \mathcal{IN}_\mathcal{O}$ be two individuals, $A, B \in \mathcal{CN}_\mathcal{O}$ be two concepts, and $r \in \mathcal{RN}_\mathcal{O}$ be a role, then the following holds:*

1. $A \sqsubseteq B \in SI_{fa}(\mathcal{O})$ iff $A \sqsubseteq B \in SC_{fa}(\mathcal{O})$;
2. $\{a\} \sqsubseteq A \in SI_{fa}(\mathcal{O})$ iff $a : A \in SC_{fa}(\mathcal{O})$;
3. $\{a\} \sqsubseteq \exists r.\{b\} \in SI_{fa}(\mathcal{O})$ iff $(a, b) : r \in SC_{fa}(\mathcal{O})$;

The theorem is similar to Theorem 12 hence we follow a similar proof. The ← direction is obvious because the additional **CR** rules are special cases of the corresponding **R** rules where certain atomic concepts are restricted to nominals.

We can prove the → directions of the theorem by proving the following stronger lemma:

Lemma 16 *For an ontology $\mathcal{O}$, let $A_{fa}(\mathcal{O}) = (\mathcal{T}, \mathcal{A}, CT, QT, IT)$ and $AI(A_{fa}(\mathcal{O}))$ be $(\mathcal{T}', CT, QT, IT)$. If we use SI to denote $SI_{fa}(\mathcal{O})$, and SC to denote $SC_{fa}(\mathcal{O})$, then the following invariants hold in each step of computation of SI, where $a, b \in \mathcal{IN}_{\mathcal{O}}$ are two individuals, $A, B \in \mathcal{CN}_{\mathcal{T}}$ are two concepts, and $r \in \mathcal{RN}_{\mathcal{O}}$ is a role:*

$$fc(A), fc(B) \in \mathcal{CN}_{\mathcal{T}} \text{ and } A \sqsubseteq B \in SI \text{ only if } A \sqsubseteq B \in SC \tag{57}$$

$$fc(A) \in \mathcal{CN}_{\mathcal{T}} \text{ and } A \sqsubseteq fc(\{a\}) \in SI \text{ only if } a : fc(A) \in SC \tag{58}$$

$$A \sqsubseteq \bot \notin SI \text{ and } A \sqsubseteq \exists r.B \in SI \text{ only if } A \sqsubseteq \exists r.B \in SC \tag{59}$$

$$A \sqsubseteq \exists r.\{a\} \text{ only if } A \sqsubseteq \exists r.\{a\} \in SC \tag{60}$$

$$\{a\} \sqsubseteq B \in SI \text{ only if } a : B \in SC \tag{61}$$

$$\{a\} \sqsubseteq \{b\} \in SI \text{ only if } a \doteq b \in SC \tag{62}$$

$$fc(\{a\}) \sqsubseteq fc(\{b\}) \in SI \text{ only if } a \doteq b \in SC \tag{63}$$

$$r \sqsubseteq s \in SI \text{ only if } r \sqsubseteq s \in SC \tag{64}$$

$$\{a\} \sqsubseteq fc(\{b\}) \in SI \text{ only if } a \not\doteq b \in SC \tag{65}$$

$$\{a\} \sqsubseteq \exists r.\{b\} \in SI \text{ only if } (a, b) : r \in SC \tag{66}$$

$$r_1 \circ r_2 \sqsubseteq s \in SI \text{ only if } r_1 \circ r_2 \sqsubseteq s \in SC \tag{67}$$

$$\{a\} \sqsubseteq \bot \notin SI \text{ and } \{a\} \sqsubseteq \exists r.B \in SI \text{ only if } a : \exists r.B \in SC \tag{68}$$

Apparently the → directions of the theorem hold if the above lemma holds. Similar as in the proof of Theorem 12, this lemma can further be proved by induction on application of rules. A detailed proof is deferred to the Appendix.

It is worth mentioning that, although the syntactic nominal-safety condition as defined in Def. 7 ensures that the optimisation of ABox reasoning presented in this section can yield the same results as the internalisation approach, it does not guarantee the completeness of ABox reasoning in general, even when the TBox classification results are complete. Below is an example:

Example 8 *Ontology $\mathcal{O}_8$ contains the following axioms:*

$$A \equiv \geq 3\, r.B \tag{69}$$

$$b_i : B \ (i = 1, 2, 3) \tag{70}$$

$$(a, b_i) : r \ (i = 1, 2, 3) \tag{71}$$

$$b_1 \not\doteq b_2 \tag{72}$$

$$b_2 \not\doteq b_3 \tag{73}$$

$$b_1 \not\doteq b_3 \tag{74}$$

It is easy to see that $\mathcal{O}_8$ satisfies the syntactic and semantic nominal safety conditions and $\mathcal{O}_8 \models a : A$. However, using the syntactic approximation techniques presented so far, $a : A$ or $\{a\} \sqsubseteq A$ will not be included in the computed closure.

The incompleteness is because our approach does not count the unique r-objects of a. Later in Sec. 8.2, we will show that supporting such kind of counting with arbitrary cardinality values will in general lead to intractability.

6.5. ABox Syntactic Approximation Summary

In this section, we presented the transformation (Def. 8) and different approximate deduction approaches, realised with different rule sets–internalisation with the **R** rules, or the **AR** rules, the **TR** rules or the **CR** rules.

This section is based on one of our previous work [66] but has been extensively revised. All definitions and reasoning rule sets have been extended to support a more expressive DL. Relation between the original ontology and the transformation result is characterised by Theorem 8. A new optimisation for transitive-only role chain is introduced (Sec. 6.3.1). A new optimisation for combined TBox and ABox reasoning with nominals is also developed (Sec. 6.4). The equivalence of results between different ABox approximate reasoning approaches is now formally characterised (Theorem 12,14 and 15). We explicitly show that the combined approach yields the same materialisation results as the internalisation approach without introducing unnecessary complements of nominals when the transformation results of the ontology is nominal-safe as specified in Theorem 6. This further indicates that they should all be tractable (Theorem 10) and soundness-guaranteed (Theorem 11).

Notably, in the combined approach (Sec. 6.4) the introduction of non-native CoNs is completely eliminated. And the introduction of non-native unpaired nominal is minimised to a single rule **CR8a**. As we indicated earlier, such introduction will not substantially increase the cost of computation or the size of the closure. This makes the combined approach the overall most preferred solution among the ones introduced in this section. When the ontology contains no nominal, this approach is as efficient as the nominal-free approach (Sec. 6.3) since it contains all the **AR** rules. When the ontology contains nominals, it is much more efficient than the internalisation approach (Sec. 6.2) due to the smaller number of terms. In the next section we will conduct experiments on ontology materialisation and compare with other off-the-shelf reasoners to show that this approach is indeed efficient and yields high recalls.

7. Evaluation

To evaluate the effectiveness of our approach and compare to other reasoners, we implemented a syntactic approximation reasoner *REL* based on the $\mathcal{EL}^{++}_{\mathcal{CQI}}$ transformation (cf. Def. 8) and the **CR** rules. Hence, its reasoning results will be the combined approximate reasoning closure (cf. Def. 13). We then conducted experiments on both TBox classification and ontology materialisation as defined in Def. 1 with the recent versions of mainstream fully-fledged OWL 2 DL reasoners. The reasoners we compared with were Pellet 2.3.0 [6], HermiT 1.3.8 [7] and FaCT++ 1.6.2 [8]. All of these reasoners and our implementation were using the same OWLAPI 3.4.5 for parsing of ontologies and retrieval of results. All experiments were conducted in an environment of 64-bit Ubuntu 14.04 with 3.20GHz CPU and 10GB RAM allocated to the JVM.

[6] `http://clarkparsia.com/pellet/`

[7] `http://www.hermit-reasoner.com/`

[8] `http://code.google.com/p/factplusplus/`

In each of the following tests, each reasoner was given a 30-minute execution time on each ontology, to create the corresponding `OWLReasoner`, to load the `OWLOntology` object created by OWLAPI, to perform reasoning, to retrieve results and then count the numbers. The recall of REL was calculated against the results of complete reasoners.

7.1. TBox Classification Evaluation

In this evaluation, we mainly examined the ontologies from the HermiT benchmark [27] and the OWL 2 DL category ontologies from the OWL Reasoner Evaluation Workshop 2013 (ORE2013) [9].

There are 393 ontologies in the HermiT Benchmark. Among these ontologies:

1. 17 ontologies do not conform to the OWL 2 standard (with malformed literals or violating the global restrictions).
2. 5 ontologies had different results from the 3 fully-fledged reasoners (when the reasoner could finishing reasoning with the ontology).
3. One ontology was identified as inconsistent by all the 4 reasoners.

We removed these ontologies from comparison as they did not have consistent and coherent classification results. This left us 370 ontologies that reasoning results from fully-fledged reasoners were coherent. Among these ontologies:

1. HermiT ran out of time on 2 ontologies and was terminated due to error on another one. It successfully classified the remaining 367 ones.
2. FaCT++ ran out of time on 6 ontologies and was terminated due to error on another 22. It successfully classified the remaining 342 ones.
3. Pellet ran out of time on 10 ontologies and ran out of memory on another one. It successfully classified the remaining 359 ones.
4. REL successfully classified all of them within 5 minutes.

The classification time of REL in comparison to that of HermiT, FaCT++ and Pellet on ontologies that could be classified by the corresponding fully-fledged reasoner is shown in Figures 1, 2 and 3, respectively. The ontologies in each figure are sorted w.r.t. their classification time by HermiT, FaCT++ and Pellet, respectively. Time unit is second in all figures.

As we can see from Figure 1-3, REL was in general more efficient than the fully-fledged reasoners when classifying the HermiT benchmark. Particularly, REL was slower than HermiT on only 12 ontologies, slower than FaCT++ on only 19 ontologies, and slower than Pellet on only 11 ontologies. And none of these ontologies took REL more than 5 seconds. On the other hand, REL was faster than fully-fledged on all ontologies that took either of them more than 1.5 seconds.

In the 370 comparable ontologies, 367 ones could be classified by at least one of the fully-fledged reasoners. On each of these ontologies, we counted the total number of ancestor and equivalent concepts for each concept (excluding $\top$ and $\bot$). We compared the numbers returned by REL and other reasoners to calculate the recall of REL. A breakdown of the recall distribution is shown in Table 11.

As we can see from the distribution. REL was empirically highly complete on the HermiT benchmark. The only one ontology with less than 98$\%$ recall contains axioms with empty conjunction and disjunction, e.g. of form $A \sqsubseteq \bigsqcap_{B \in \emptyset} B$. After removing such axioms, REL could

[9] ORE workshop website: `http://ore2013.cs.manchester.ac.uk/`

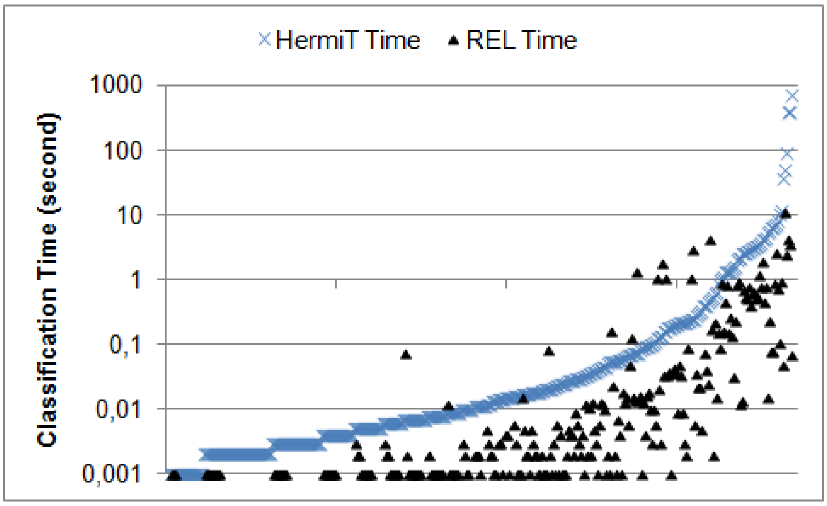

Figure 1: Classification Time of REL and HermiT on HermiT Benchmark

Table 11: Recall Distribution of REL on HermiT Benchmark

recall	#ontology	%ontology
100%	358	97.55%
98-100%	8	2.18%
<98%	1	0.27%

achieve 100% recall. All the ontologies that REL was incomplete are relatively easy. They could be classified by HermiT in less than 8 seconds. REL yielded high recall on ontologies that were hard for fully-fledged reasoners. In Table 12 we show the time of all reasoners and recall of REL on ontologies that none of the fully-fledged reasoners could classify in 10 seconds. Note that there were 2 almost identical variants of FMA Lite. In the table we show the average classification time on these 2 ontologies. As we can see in the table, apart from the Gazetteer ontology, these difficult ontologies are not in tractable DLs. In our additional study, we have also confirmed that syntactic approximation can achieve 100% recall on the FMA Constitutional ontology [60].

Table 12: Comparison on Hard Ontologies in HermiT Benchmark (t/o: time out; N/A: not applicable)

Ontology $\mathcal{O}$	DL	HermiT	FaCT++	Pellet	REL	
					time	recall
FMA Constitutional	$\mathcal{ALCOIF}(\mathcal{D})$	t/o	t/o	t/o	58.91	N/A
NCI Thesaurus	$\mathcal{ALCH}(\mathcal{D})$	t/o	t/o	t/o	274.583	N/A
Full Galen from Web	$\mathcal{ELHIF}^+$	error	t/o	t/o	44.619	N/A
FMA Lite	$\mathcal{ELI}^+$	385.072	t/o	t/o	3.983	100%
Gazetteer	$\mathcal{EL}^+$	51.131	t/o	29.881	10.874	100%

The OWL 2 DL category of the ORE2013 benchmark was used for TBox classification com-

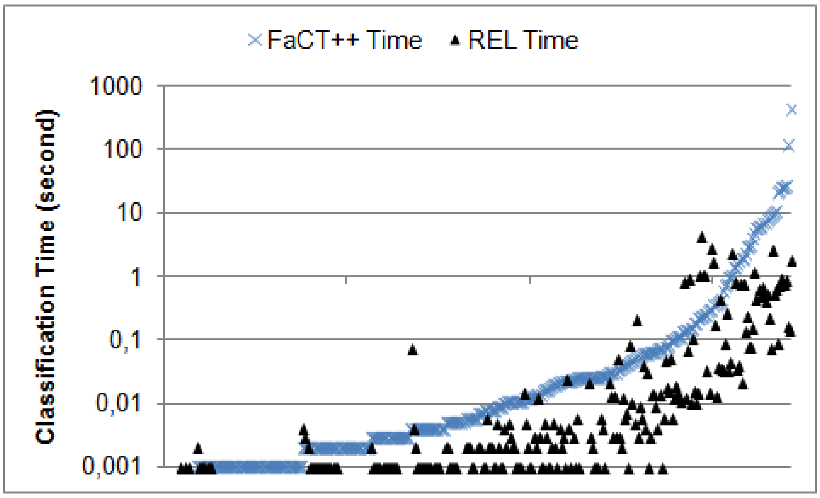

Figure 2: Classification Time of REL and FaCT++ on HermiT Benchmark

petition, in which HermiT was the winner. Hence, we classified these ontologies with both HermiT and REL. Same as before, HermiT and REL were given 30 minutes to classify each of the 203 ontologies. Among these ontologies, HermiT managed to finish reasoning 191 of them without timeout. REL managed to finish reasoning all of them without timeout. In the ontologies finished by HermiT, 5 were inconsistent. REL was able to identify 2 of them. The classification time of REL and Hermit on the 186 ontologies that both reasoners classified without inconsistency are illustrated in Figure 4. The ontologies are sorted by their classification time of HermiT. The time unit is second.

As one can see from the figure, REL was in general faster than HermiT. Out of the 186 ontologies, REL was faster on 162 of them, and took between 100% to 500% time of HermiT on another 21 of them, took more than 10 times time on only 2 of them. In total, REL classification time was about 13.55% of HermiT classification time on these ontologies. REL was notably faster than HermiT on ontologies that were difficult for HermiT (on the right end of Figure 4). Furthermore, on the 12 ontologies that HermiT ran out of time, REL spent at most 22.71 seconds and in average 5.80 seconds.

The distribution of recall of REL on the 186 ontologies is shown in Table 13.

Table 13: Recall Distribution of REL on ORE2013 Benchmark

recall	#ontology	%ontology
100%	148	79.57%
95-100%	30	16.13%
80-95%	1	0.54%
70-80%	2	1.08%
60-70%	1	0.54%
50-60%	1	0.54%
<50%	3	1.61%

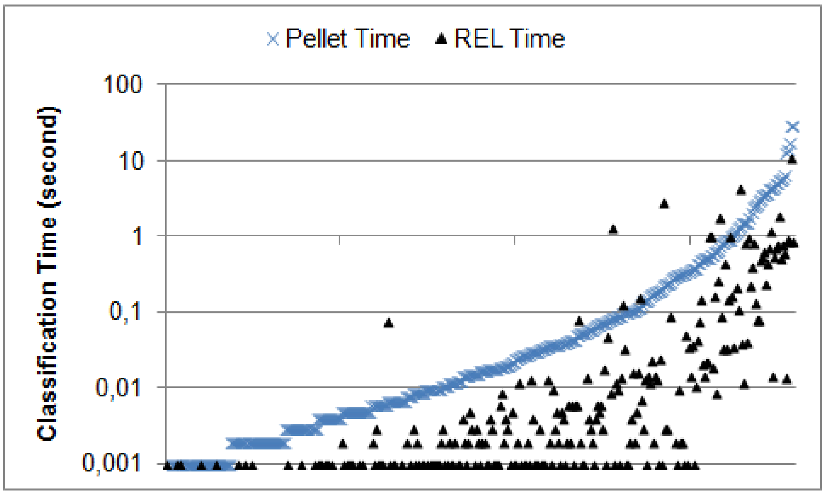

Figure 3: Classification Time of REL and Pellet on HermiT Benchmark

As one can see, REL was complete on 79.57% of the ontologies, was rather complete (recall $\geq 95\%$) on most of the remaining ontologies. The ontologies that REL was incomplete are relatively simple ontologies. They could be classified by HermiT within 10 seconds.

To better understand the incompleteness of REL, we also looked into the ontologies with incomplete results and analysed the causes for incompleteness:

1. All of the 4 ontologies with recall lower than 60% are caused by disjointness between datatypes, such as `literal` and `long`. Since datatypes were not discussed in this paper and were not fully supported in the evaluated REL, the reasoner failed to recognise such disjointness, and hence failed to infer the unsatisfiability of many concepts. This significantly reduced the recall of REL. Nevertheless, such a problem can be easily resolved by treating datatypes as concepts and asserting the disjointness among them. By doing so REL could achieve 100% recall on these ontologies.
2. Among the other incomplete ontologies, 14 of them are variants of the Pizza ontology. [10] This ontology contains cardinality restrictions in axioms such as:

$$InterestingPizza \equiv Pizza \sqcap (\geq 3\ hasTopping.\top)$$

   This axiom is similar to the axiom (69) in Example 8. In order to infer that a given $Pizza$ is an $InterestingPizza$, the reasoner needs to identify 3 mutually disjoint $hasTopping$-fillers of the concept. This can be realised by the following rule:

   If $X \sqsubseteq \exists r.B_1, \ldots, X \sqsubseteq \exists r.B_n, B_i \sqsubseteq B, B_i \sqsubseteq fc(B_j), r \sqsubseteq s$ and $(Y, B, s, n) \in QT$, then $X \sqsubseteq Y$

   In general, this rule is intractable, as we will discuss in more detail in Sec. 8.2. Nevertheless, when the maximal value of n in this rule is a constant, the above rule is tractable. For

[10] Pizza ontology: `http://130.88.198.11/co-ode-files/ontologies/pizza.owl`

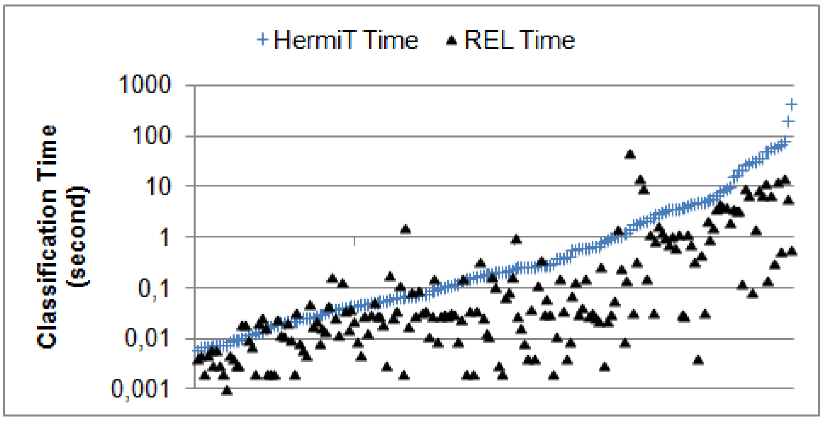

Figure 4: Classification Time of REL and HermiT on ORE2013 Benchmark

the pizza ontology variants, $n \leq 3$. Hence, this rule can be executed in a tractable manner. By including this rule in the rule set of REL, REL yields 100% results on all the 14 pizza variants.

3. Another pattern similar as the above one involves universal restriction with an enumeration as filler. For example, if there are following axioms in the ontology:

$$X \sqsubseteq \forall r.\{a_1, \ldots, a_n\}$$
$$a_i : A$$
$$Y \equiv \leq k\, r.A$$

then it is possible to infer that $X \sqsubseteq Y$ if there are at least $(n - k)$ pairs of equivalent individuals in $a_1, \ldots, a_n$. With syntactic approximation, such a pattern can be resolved using the same cardinality restriction checking rule as shown before. But it will leads to intractability in the same way in general case if the maximal cardinality value is not fixed.

Such observation shows that REL is in general efficient and rather complete on the majority of ontologies in the ORE2013 benchmark. The recall of REL can be further improved without affecting its tractability by considering more tailored rules based on the axiom patterns in ontologies.

To give an indication about why it is non-trivial to perform the syntactic approximation, we also used a pure $\mathcal{EL}^{++}$ reasoner ELK 0.4.1 [11] to classify the ontologies tested in the TBox classification evaluation and investigated its recall. Although ELK was rather efficient on these ontologies, it ignored all axioms with expressive power beyond $\mathcal{EL}^{++}$. This led to rather incomplete results on some complex ontologies. For example, FMA has 33,433 unsatisfiable concepts [60]. ELK was not able to find any of them except the $\bot$. Hence, its recall was as low as 0.4% on this ontology. Such a phenomena can also be observed on the ORE benchmark. The distribution of recall of ELK on the same 186 ontologies that REL, HermiT and ELK could classify is presented in Table 14. Comparing Table 14 to Table 13, it can be seen that ELK is much less complete than REL on these ontologies and the additional rules implemented in REL helped improve the recall.

[11] ELK reasoner website: `https://code.google.com/p/elk-reasoner/`

Table 14: Recall Distribution of ELK on ORE2013 Benchmark

recall	#ontology	%ontology
100%	113	60.75%
95-100%	12	6.45%
80-95%	20	10.75%
70-80%	19	10.22%
60-70%	3	1.61%
50-60%	6	3.23%
<50%	13	6.99%

The comparison between recalls of REL and ELK clearly suggests that using only an $\mathcal{EL}^{++}$ algorithm to reason with ontologies in more expressive DLs will often leads to significantly incomplete results. The approximate deduction rules utilised in syntactic approximation can exploit the semantics of the non-$\mathcal{EL}^{++}$ axioms in these ontologies to partially fill the incompleteness gap and yield high recall on these ontologies.

7.2. Ontology Materialisation Evaluation

In this evaluation, we created a test suite using both benchmark and real-world ontologies. We included all ontologies in the HermiT Benchmark with more than 100 individuals. We also included the SEMINTEC [12] ontology, which is developed for semantic web mining, and the VICODI [13] ontology, which is developed to represent the history of Europe. Both of them have been used as benchmark for ABox reasoning [58]. Besides, we used the ADOxx Metamodelling ontologies provided by the BoC-group. [14] These ontologies followed the research in the MOST (Marrying Ontology and Software Technologies) project, [15] using ontologies to represent meta-modelling information in model-driven software development and using reasoning to detect inconsistencies. We used 3 such ontologies with identical TBox and varying size of ABoxes. [16] We also used the Travel ontology (v26) from the SWAT (Semantic Web Authoring Tool) project. [17] Both the BoC ontologies and the Travel ontology are in highly expressive DLs and have complex TBox and ABox. Some of the ontologies have imported other ontologies (such as Travel). We have merged these ontologies with their imported ontologies so the materialisation results presented later are all for the import closures.

For each ontology, we performed ontology materialisation, getting all the subsumptions between named concepts, all named classifications of all individuals and all named relations between all pairs of individuals. Recall was then calculated for each of these 3 categories of results.

There are 46 ontologies in the HermiT Benchmark with more than 100 individuals. Among these ontologies, there are 2 ontologies that the fully-fledged reasoners yielded different results (when they were able to finish materialising). Another ontology was identified inconsistent by all 4 reasoners. This left us 43 materialisable ontologies that the fully-fledged reasoners did not have conflicting results. In these ontologies:

[12] http://www.cs.put.poznan.pl/alawrynowicz/semintec.htm

[13] http://www.vicodi.org/about.htm

[14] http://www.boc-group.com/

[15] http://217.74.68.230/

[16] Due to intellectual property rights reasons, we have anonymised these ontologies.

[17] http://swatproject.org/

1. HermiT ran out of time on one of them and finished materialisation on the remaining 42.
2. FaCT++ ran out of time on 3 of them and was terminated with error on the other 4. It successfully materialised 36 ontologies.
3. Pellet ran out of time on one of them and was terminated with error on another. It successfully materialised 41 ontologies.
4. REL finished materialising all of them. The hardest one took REL about 3 seconds.

The materialisation time of REL in comparison to that of the fully-fledged reasoners on their corresponding materialisable ontologies is shown in Figure 5. The ontologies are sorted w.r.t. their materialisation time by the fully-fledged reasoners. Time unit is second.

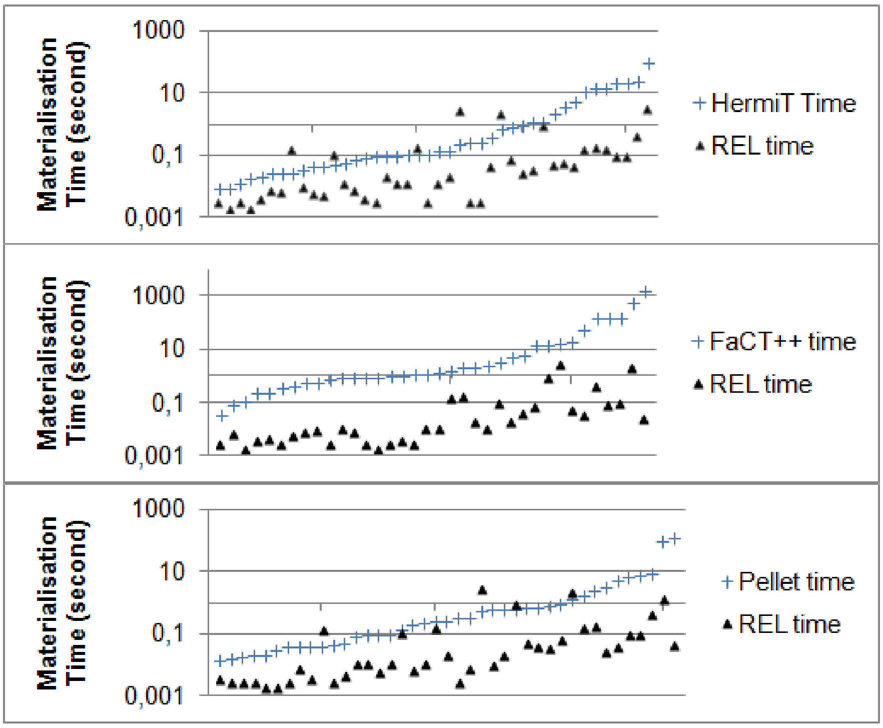

Figure 5: Materialisation Time on HermiT Benchmark

As we can see from Figure 5, REL was in general more efficient than the other reasoners on the materialised ontologies. Particularly, REL was slower than HermiT on 5 ontologies and slower than Pellet on 4 ontologies. None of these ontologies took REL more than 3 seconds. On the other hand, REL was more efficient than HermiT or Pellet on any ontology that took them more than 2 seconds and was at least 4 times faster than FaCT++ on all ontologies.

The recalls of REL on entailed named concept subsumptions, named concept assertions and named role assertions were calculated for all ontologies that at least one fully-fledged reasoner could materialise. The distribution of these recalls is shown in Table 15

As we can see from Table 15, the recall of REL is practically high on materialising the HermiT Benchmark. Subsumption and role assertion recalls are 100%. Class assertion is incomplete

Table 15: REL Recall on HermiT Benchmark (Sub.: subsumption; CA.: class assertion; RA.: role assertion)

recall	Sub.		CA.		RA.	
	#ontology	%ontology	#ontology	%ontology	#ontology	%ontology
100%	43	100%	41	95.35%	43	100%
99-100%	0	0%	1	2.33%	0	0%
95-99%	0	0%	1	2.33%	0	0%

on only 2 ontologies, and the recalls are both above 95%. Similar as in the TBox classification, the two ontologies on which REL was incomplete are not very difficult. Both of them could be materialised by HermiT within one second. While REL performed well on ontologies difficult for other reasoners. In Table 16 we show the time of all reasoners and recall of REL on ontologies that none of the fully-fledged reasoners could materialise in 5 seconds. Note that, there were 3 almost identical variants of the Wine/Food ontology. In the table we show the average time and recall on these 3 ontologies for Wine/Food. Also, the 100% recall of REL means that REL had all the results that were yielded by the fully-fledged reasoner.

Table 16: Materialisation Comparison on Hard Ontologies in HermiT Benchmark (t/o: time out)

Ontology $\mathcal{O}$	DL	HermiT	FaCT++	Pellet	REL	
					time	recall
MGED	$\mathcal{ELFO}(\mathcal{D})$	5.55	error	5.239	0.04	100%
Sweet Phenomena	$\mathcal{SHOIN}(\mathcal{D})$	14.65	error	error	0.15	100%
Wine/Food	$\mathcal{SHOIN}(\mathcal{D})$	22.59	147.62	8.00	0.21	100%
OBI	$\mathcal{SHOIN}(\mathcal{D})$	101.21	t/o	t/o	2.98	100%
Teleost taxonomy	$\mathcal{EL}$	t/o	t/o	95.04	1.42	100%

The results shown in Table 16 suggest that REL was efficient and yielded all intended results when materialising the hardest ontologies in the HermiT Benchmark. It is worth mentioning that such difficult ontologies are not necessarily big. For example, the Wine/Food ontology contains only 138 concepts, 16 roles, 206 individuals and 889 axioms. But it was still not easy for fully-fledged reasoners to materialise. Also note that apart from the Teleost taxonomy, the other ontologies are not in tractable DLs. Hence, the efficient and highly complete results of REL on these ontologies could not have been achieved without approximation.

The additional ontologies we used for materialisation evaluation are shown in Table 17. Similar as before, every reasoner was given 30 minutes to materialise each ontology. No ontology had different results from the fully-fledged reasoners (when they were able to materialise). Recall of REL was calculated by comparing results of REL and the fully-fledged reasoners. The time of all reasoners and recall of REL on these ontologies are shown in Table 18. Time unit is second.

From the results in Table 18 we can see that in general REL preformed very well on materialisation of these ontologies, especially when the ontology was hard for fully-fledged reasoners. Notably, REL had a 100% recall on all ontologies except the Travel, even if some of the ontologies are in very expressive DL and are quite complex. This means that REL had 100% results for named concept subsumption, named class assertion and named role assertion, respectively.

We were, however, unable to obtain the accurate recall of REL on the Travle ontology. This is because HermiT, as the only fully-fledged reasoner able to materialise the Travel ontology, did

Table 17: Additional Ontology Materialisation Benchmark Ontologies

Ontology $\mathcal{O}$	DL	$\lvert\mathcal{CN}\rvert$	$\lvert\mathcal{RN}\rvert$	$\lvert\mathcal{IN}\rvert$	$\lvert\mathcal{O}\rvert$
Semintec	$\mathcal{ALCIF}$	60	16	17941	65459
VICODI	$\mathcal{ALUH}$	184	10	29614	114375
BoC small	$\mathcal{SROIQ}$	408	46	387	4616
BoC middle	$\mathcal{SROIQ}$	408	46	2067	25435
BoC big	$\mathcal{SROIQ}$	408	46	7734	99525
Travel	$\mathcal{SROIQ(D)}$	269	150	2032	8507

Table 18: Additional Ontology Materialisation Comparison (t/o: out of time)

Ontology $\mathcal{O}$	HermiT	FaCT++	Pellet	REL	
				time	recall
Semintec	1.317	t/o	1.764	2.09	100%
VICODI	6.283	t/o	4.041	1.655	100%
BoC small	12.634	271.439	40.365	0.714	100%
BoC middle	175.674	t/o	t/o	3.065	100%
BoC big	1302.235	t/o	t/o	16.746	100%
Travel	1484.948	t/o	t/o	7.834	<100%

not yield all materialisation results. For example, Travel ontology contains the following axioms:

$$is\_part\_of \equiv has\_part^{-} \tag{75}$$

$$EuropeanIsland \equiv Island \sqcap \exists is\_part\_of.\{europe\} \tag{76}$$

In addition, HermiT was able to entail that:

$$(europe, oldany\_island) : has\_part \tag{77}$$

$$oldany\_island : Island \tag{78}$$

It should have further entail the following two results but it did not:

$$(oldany\_island, europe) : is\_part\_of \tag{79}$$

$$oldany\_island : EuropeanIsland \tag{80}$$

Hence, the results yielded by HermiT were incomplete and we cannot know what the complete results should be. We did notice that the results of REL were also incomplete. By looking into the details of this ontology we realised that Travel is actually not a nominal-safe ontology. It contains axioms like the following

$$isola\_comacina : \exists directPartOf.(\{continental\_europe\} \sqcap \exists is\_occupied\_by.\{italy\})$$

Assuming $fa(\{continental\_europe\} \sqcap \exists is\_occupied\_by.\{italy\}) = X$, the above axiom will lead to an $\mathcal{EL}^{++}_{\mathcal{CQI}}$ transformation including the following axiom

$$X \sqsubseteq \{continental\_europe\},$$

which is not a nominal-safe $\mathcal{EL}^{++}$ axiom. Hence the relation $(continental\_europe, italy)$: $is\_occupied\_by$ cannot be inferred with the current rule set in REL. We have identified other similar axioms in both TBox and ABox of Travel that are not nominal-safe, which partially explains the missing of some results and also motivates us to investigate approximation based on a complete $\mathcal{EL}^{++}$ rule set in the future.

7.3. Entailments with Unsafe Nominals in Evaluation Benchmark

In the evaluation presented in the previous two sections, we have tested ontologies with different levels of expressive powers, including ontologies without nominals, with only safe nominals as defined in Def. 7, and potentially with unsafe nominals. As we have shown in Theorem 12 and Theorem 15, the rule set implemented in the evaluated REL should yield as many results as an internalisation-based approach (as presented in Sec. 6.2) on nominal-free and nominal-safe ontologies. In order to have a better overview of nominal-free and nominal-safe ontologies in our evaluation benchmark. We obtained statistics of unsafe entailments (entailments with unsafe nominals) computed by REL in the evaluation benchmark.

In the 585 ontologies we used, there are 326 ones with at least one individual. The others apparently will not cause any unsafe entailment. In the 326 ontologies with individuals, there are 242 ones without any unsafe entailment. These ontologies are either nominal free, or they satisfy the nominal-safety condition. Some of the hardest ontologies for fully-fledged reasoners, such as FMA Constitutional, are included in these 242 ones. In the other 84 ontologies with unsafe entailments, such entailments are few and constitute only a small percentage of the total entailments inferred by REL. Only 28 ontologies have more than 10 unsafe entailments. Only one ontology has more than 200 unsafe entailments. The one with the most unsafe entailments is GALEN-Full-Union_ALCHOI(D), on which REL inferred 635 unsafe entailments. This ontology, however, has a closure as large as 1,656,159 entailments, in which unsafe ones constitute only 0.04%. Only 6 ontologies have more than 1% unsafe entailments in their closure. The highest percentage of unsafe entailments is 4.23% in these ontologies.

These statistics suggest that the nominal-free and nominal-safety conditions can be satisfied by a large number of real-world ontologies, including some of the most difficult ones such as FMA Constitutional. Even when it is not satisfied, an ontology usually contains only a very small percentage of unsafe entailments in its closure. Using syntactic approximation on these ontologies may lead to incomplete results. But as our evaluation shows, in most cases, the recall is empirically high.

8. Discussion

In this section, we discuss the relation between our approach and other related works. After that, we will analyse in more detail the incompleteness of our approach and present the idea and first result of a partial completeness guarantee of our approach.

8.1. Related Work

Our approach is related to and different from several existing works:

1. As shown in the previous sections, the reasoning of our approach is achieved by computing the closure of some completion rule set that includes the original $\mathcal{EL}^{++}$ completion-rules and additional rules that exploit the non-$\mathcal{EL}^{++}$ features. Hence, our approach can be regarded as a *consequence-based* approach.

Our rule sets include the rules presented in Table 2 and therefore it is sound and complete for nominal-safe $\mathcal{EL}^{++}$. A difference is that, the transformation presented in Def. 4 introduces a pair $(fa(C), fa(\sim C))$ into CT for each term C in the ontology, even if C is an $\mathcal{EL}^{++}$ concept, and such a structure will be exploited by the additional rules. Hence, the closure computed by our approach will be larger than that computed by the original $\mathcal{EL}^{++}$ rules even when applied on $\mathcal{EL}^{++}$ ontologies. For example, if the original ontology contains an axiom $C \sqsubseteq D$, then due to the involvement of rule **R10**, the closure of our approach will include $fa(\sim D) \sqsubseteq fa(\sim C)$, which corresponds to $\neg D \sqsubseteq \neg C$ in the original ontology. Such unnecessary entailments in the closure will not increase the complexity of reasoning but will make our approach less efficient than dedicated $\mathcal{EL}^{++}$ reasoners on pure $\mathcal{EL}^{++}$ ontologies.

Similar to other consequence-based procedures, the consequences of executing a completion rule can be used to identify the next rules to be executed. Due to the fact that the consequences of our rules are always normal form $\mathcal{EL}^{++}$ axioms, our rule set can be implemented with a rule-chaining mechanism similar to the one presented by Baader et al. [4]. A difference from other consequence-based procedures is that, our rule sets include rules whose premise and consequence do not share any concept or role. For example in **R10**, the premise is $A \sqsubseteq B$, while the consequence is $fc(B) \sqsubseteq fc(A)$. Although A and $fc(A)$ (B and $fc(B)$) are related in the CT, they are not structurally or syntactically related in the closure. To incorporate such rules, the $\mathcal{EL}^{++}$ due-ontology classification mechanism [77] can be applied so that the completion procedure can be treated as incremental reasoning with the consequences of these rules as newly added axioms. The REL reasoner we evaluated in Sec. 7 was implemented with such mechanisms.

2. The idea of approximating the reasoning of a complex DL with the reasoning of a simpler DL is related to that of *knowledge compilation*. More specifically, a closure S computed by our approach can be regarded as an upper bound of the original ontology $\mathcal{O}$ in the sense that for each $\alpha \in S$, if the signature of α is included in that of $\mathcal{O}$, then $\mathcal{O} \models \alpha$ (Theorem 5). Nevertheless, there are several essential differences. First of all, knowledge compilation [38, 39, 71, 72] and its DL adoptions [61, 6] attemp to compute the least upper bounds (and the greatest lower bound) while our approach does not guarantee to compute the least upper bound. We argue that this should not be regarded as a limitation of our approach. Instead, it is an inevitable consequence of the high complexity of finding a least upper bounds. In fact, finding the Horn least upper bound of a propositional theory is already NP-hard [39]. Finding a tractabel-DL least upper bound for an intractabel-DL ontology is not easier. Particularly, for an ontology $\mathcal{O}$ in a DL whose classification cannot be performed in polynomial time, if one can compute in polynomial time a $\mathcal{EL}^{++}$ least upper bound $\mathcal{O}'$ of $\mathcal{O}$ s.t. for any other upper bound $\mathcal{O}''$ we have $\mathcal{O}' \models \mathcal{O}''$, then the classification of $\mathcal{O}$ can be performed by classifying $\mathcal{O}'$ because $\mathcal{O}'' = \{A \sqsubseteq B | \mathcal{O} \models A \sqsubseteq B\}$ is also an upper bound of $\mathcal{O}$. This is contradictory to the intractability of classification of $\mathcal{O}$. This implies that, it is impossible to find the semantically most complete $\mathcal{EL}^{++}$ approximation of the original ontology in polynomial time. Without attempting to compute such an optimal approximation, our approach, as Theorem 4 and 10 showed, is tractable.

   The NP-hardness of computing the least upper bound also leads to the separation of off-line and on-line computation in knowledge compilation. In order to achieve tractable reasoning in the on-line stage, intractable compilation is performed off line and can be time-consuming. Our approach presented in this paper does not require such an off-line compilation stage and its entire procedure is tractable.

3. Similar as the original $\mathcal{EL}^{++}$ rules, the additional completion rules presented in this paper are individually tractable. This is achieved by ensuring that all the rules can be executed in polynomial time and the size of the closure is polynomial w.r.t. the size of the original ontology. This is different from some other intractable consequence-based algorithm [41, 73], in which the execution of certain rules may lead to an exponential blowup of the closure. The idea of avoiding complex inference steps is similar to that of approximate deduction. For example, disallowing the *modus ponens* rule in propositional calculus facilitates tractable consistency checking [23, 22]. Despite the conceptual resemblance, the application setting and concrete techniques of our approach is quite different from these earlier works. Most notably, we are dealing with DLs instead of propositional logic.
   Of course, avoiding intractable rule execution will affect the completeness of our rule sets on expressive DLs. In the next section, we will further discuss such impact in more detail with examples.

8.2. Incompleteness

Our approach is soundness guaranteed but incomplete. Due to the intractability of $\mathcal{SROIQ}$, it is in general not possible to develop a tractable procedure that guarantees complete results for $\mathcal{SROIQ}$. In fact, the simple combination of $\mathcal{EL}^{++}$ with inverse role will cause intractability [1]. In this section, we show that repairing certain incompleteness of our approach will lead to an intractable procedure.

The first type of such incompleteness is related to resolution. Consider the following example:

Example 9 *Ontology $\mathcal{O}_9$ contains the following axioms:*

$$A \sqsubseteq B \sqcup C \tag{81}$$
$$A \sqsubseteq D \sqcup \neg C \tag{82}$$
$$B \sqcup D \sqsubseteq E \tag{83}$$

A name assignment $fa(\textbf{.})$ assigns the following names (among others):

$$\begin{aligned} fa(B \sqcup C) &= X_1 \\ fa(\neg B \sqcap \neg C) &= cX_1 \\ fa(D \sqcup \neg C) &= X_2 \\ fa(\neg D \sqcap C) &= cX_2 \\ fa(B \sqcup D) &= X_3 \\ fa(\neg B \sqcap \neg D) &= cX_3 \\ fa(\neg A) &= cA \\ fa(\neg B) &= cB \\ fa(\neg C) &= cC \\ fa(\neg D) &= cD \end{aligned}$$

Then we shall have $A_{fa}(\mathcal{O}_9) = (\mathcal{T}_9, \mathcal{A}_9, CT_9, QT_9, IT_9)$, where $\mathcal{T}_9$ contains the following axioms:

$$
\begin{align}
A &\sqsubseteq X_1 \tag{84}\\
A &\sqsubseteq X_2 \tag{85}\\
X_3 &\sqsubseteq E \tag{86}\\
cX_1 &\equiv cB \sqcap cC \tag{87}\\
cX_2 &\equiv cD \sqcap C \tag{88}\\
cX_3 &\equiv cB \sqcap cD \tag{89}
\end{align}
$$

CT_9 contains pairs such as (X_i, cX_i) $(i = 1, 2, 3)$, (A, cA), (B, cB), (C, cC), (D, cD).

Obviously, we have $\mathcal{O}_9 \models A \sqsubseteq E$. However, such a result cannot be computed by our approach proposed in this paper. This is because our approach does not resolve conjunctive disjunctions, e.g. resolving $(B \sqcup C) \sqcap (D \sqcup \neg C)$ into $B \sqcup D$.

One can, for example, extend the rule set with the following resolution rule:

If $A_1 \sqcap \cdots \sqcap A_i \sqcap \ldots A_n \sqsubseteq X_1, B_1 \sqcap \cdots \sqcap B_j \sqcap \ldots B_m \sqsubseteq X_2, X_1 \sqsubseteq Y, X_2 \sqsubseteq Y, A_i \sqsubseteq fc(B_j)$,
then $A_1 \sqcap \cdots \sqcap A_{i-1} \sqcap A_{i+1} \sqcap \cdots \sqcap A_n \sqcap B_1 \sqcap \cdots \sqcap B_{j-1} \sqcap B_{j+1} \sqcap \cdots \sqcap B_m \sqsubseteq Y$.

Then the rule set will be able to infer that $A \sqsubseteq E$. Particularly, from (84) and (85) one can infer that $cX_1 \sqsubseteq cA$ and $cX_2 \sqsubseteq cA$. Then applying the above resolution rule together with (87) and (88) yields that $cB \sqcap cD \sqsubseteq cA$. Together with (89) this implies that $cX_3 \sqsubseteq cA$, which can be further combined with (86) to infer that $A \sqsubseteq E$.

However, such a new rule will generate consequences of form $A_1 \sqcap \ldots A_n \sqsubseteq X$, where A_i are atomic concepts in the approximated TBox and n is bounded by the size of the vocabulary. In worst case, the total number of such axioms will be exponential w.r.t. the size of the ontology. This means that the inclusion of such a rule will lead to intractability of the completion closure. In fact, consequence-based algorithm for $\mathcal{ALCH}$ [73] includes a variation of the above resolution rule (the $\mathbf{R}^n_\sqcap$ rule) and consequently such an algorithm is intractable.

This intractability result is not surprising. It has been shown that an exponential lower bound can be established for the time complexity of resolution [82]. Therefore, it is impossible to develop a tractable algorithm that supports resolution in general. In order to provide worst-case tractable classification and materialisation, our approach does not fully cover resolution and is incomplete in this regard.

Another type of incompleteness is related to cardinality. In Example 8 we already discussed such kind of incompleteness. It is also the reason behind the incompleteness of REL on the Pizza ontology variants in our evaluation. Below is an example showing a more generic case:

Example 10 *Ontology $\mathcal{O}_{10}$ contains the following axioms (among others):*

$$X \sqsubseteq \exists r_i.B_i \ (i = 1, \ldots, k) \tag{90}$$

$$B_i \sqsubseteq B \ (i = 1, \ldots, k) \tag{91}$$

$$r_i \sqsubseteq r \ (i = 1, \ldots, k) \tag{92}$$

$$X \sqsubseteq \geq N_j \ r_j.B_j \ (j = k+1, \ldots, n) \tag{93}$$

$$B_j \sqsubseteq B \ (j = k+1, \ldots, n) \tag{94}$$

$$r_j \sqsubseteq r \ (j = k+1, \ldots, n) \tag{95}$$

$$\geq N \ r.B \sqsubseteq Y \tag{96}$$

And there is a set $\mathcal{B}_1 \subseteq \{B_1, \ldots, B_k\}$ and another set $\mathcal{B}_2 \subseteq \{B_{k+1}, \ldots, B_n\}$ s.t. the following holds:

$$\forall B_i, B_j \in \mathcal{B}_1 \cup \mathcal{B}_2, \textit{there is } B_i \sqsubseteq \neg B_j \in \mathcal{O}_{10} \tag{97}$$

$$|\mathcal{B}_1| + \Sigma_{B_j \in \mathcal{B}_2} N_j \geq N \tag{98}$$

A name assignment $fa(.)$ assigns names such that

$$\begin{aligned} fa(\geq N_j \ r_j.B_j) &= Y_j \ (j = k+1, \ldots, n) \\ fa(\geq N \ r.B) &= Z \\ fa(\neg B_i) &= cB_i \ (i = 1, \ldots, n) \end{aligned}$$

Then we shall have $A_{fa}(\mathcal{O}_{10}) = (\mathcal{T}_{10}, \mathcal{A}_{10}, CT_{10}, QT_{10}, IT_{10})$. Particularly, the above axioms (93) will be approximated into $X \sqsubseteq Y_j \in \mathcal{T}_{10}$, where $(Y_j, B_j, r_j, N_j) \in QT_{10}$. The above axiom (96) will be approximated into $Z \sqsubseteq Y \in \mathcal{T}_{10}$, where $(Z, B, r, N) \in QT_{10}$. And the above axioms in (97) will be approximated into $(B_i \sqsubseteq cB_j) \in \mathcal{T}_{10}$, where $(B_j, cB_j) \in CT_{10}$.

It can be shown that $\mathcal{O}_{10} \models X \sqsubseteq Y$:

1. Let x be an arbitrary instance of X, x will have an r_i-object of type B_i for each $B_i \in \mathcal{B}_1$, due to axioms (90).
2. Similarly, x will have N_j r_j-objects of type B_j for each $B_j \in \mathcal{B}_2$, due to axioms (93).
3. Since $r_{i/j} \sqsubseteq r$ and $B_{i/j} \sqsubseteq B$, x will have $|\mathcal{B}_1| + \Sigma_{B_j \in \mathcal{B}_2} N_j$ r-objects.
4. Due to condition (97), all these r-objects are different from each other.
5. Due to condition (98), x is of type $\geq N \ r.B$, hence x is also an instance of Y.

However, it can be realised that $X \sqsubseteq Y$ will not be computed by our approach presented in this paper. This is because our approach does not count the unique fillers of existential and minimal cardinality restrictions.

Such incompleteness can be repaired by the following cardinality counting rule:

If $X \sqsubseteq \exists s.A_i \ (i = 1, \ldots, l), X \sqsubseteq W_j$ with $(W_j, A_j, s_j, M_j) \in QT \ (j = l+1, \ldots, m)$,
$A_i \sqsubseteq A \ (i = 1, \ldots, m), s_j \sqsubseteq s \ (j = l+1, \ldots, m), A_i \sqsubseteq fc(A_j) \ (i, j = 1, \ldots, m)$,
$l + \Sigma_{j=l+1,\ldots,m} M_j \geq M$ and $(W, A, s, M) \in QT$, then $X \sqsubseteq W$

Our approach can infer that $X \sqsubseteq Y$ with the extended rule set.

Nevertheless, such a new rule will also introduce intractability into the reasoning procedure. Although the rule does not generate new form of consequence, its premise has a larger search space. Particularly, there are in total k B_is and $(n-k)$ Y_js in $A_{fa}(\mathcal{O}_{10})$. In order to execute the rule for $A_{fa}(\mathcal{O}_{10})$, the algorithm needs to find l B_is from the total k ones and $(m-l)$ Y_js (and correspondingly the B_js) from the total $(n-k)$ ones that satisfy the cardinality restriction and disjointness. Note that $1 \leq m \leq n$ is a variable. In worst case, it will need to search for 2^n combinations of B_is and B_js. Even if we disallow the use of axioms similar to (93) in the original ontology, the algorithm will still need to search for $\frac{k!}{N!}$ combinations of mutually-disjoint B_is. If the ontology contains multiple cardinality restriction axioms of form (96) with different values of N, in worst case, the algorithm still needs to perform 2^k searches, where k is bounded by the size of the concept vocabulary.

In order to avoid intractability in our approach, we did not include the above cardinality counting rule in the rule set presented in this paper.

Nevertheless, when the maximal value of cardinality in all cardinality restrictions is a constant, the above rule is tractable. Such a phenomenon is exploited by our TrOWL reasoner [18] to provide efficient reasoning in ontology-based pharmacogenomic decision support solution with the Genomic-CDS ontology [54].

A further type of incompleteness is caused by the interaction between existential restrictions and universal restrictions. Below is a simple example illustrating such an interaction:

Example 11 *Ontology $\mathcal{O}_{11}$ contains the following axioms:*

$$X \sqsubseteq \exists r.A \tag{99}$$

$$\exists r^-.X \sqsubseteq B \tag{100}$$

$$A \sqcap B \sqsubseteq C \tag{101}$$

$$\exists r.C \sqsubseteq D \tag{102}$$

A name assignment $fa(\cdot)$ assigns the following names:

$$fa(r^-) = ir$$
$$fa(\exists r^-.X) = Y$$

Then we shall have $A_{fa}(\mathcal{O}_{11}) = (\mathcal{T}_{11}, \mathcal{A}_{11}, CT_{11}, QT_{11}, IT_{11})$. Particularly, the above axiom (100) will be approximated to $Y \sqsubseteq B$ and $Y \equiv \exists ir.X$ in $\mathcal{T}_{11}$. IT_{11} will include (r, ir).

From the example it can be derived that $\mathcal{O}_{11} \models X \sqsubseteq D$. Particularly, axioms (100) is equivalent to $X \sqsubseteq \forall r.B$, which together with axiom (99) entails the following

$$X \sqsubseteq \exists r.(A \sqcap B) \tag{103}$$

This entailment is exactly what is missing from our approach, which makes it not possible to derive $X \sqsubseteq D$ with our approach. It can be recovered if the rule set is extended with the following rule, where C is of form $A_1 \sqcap \cdots \sqcap A_n$ $(n \geq 1)$:

$$\text{If } X \sqsubseteq \exists r.C, Y \sqsubseteq B, Y \equiv \exists fi(r).X, \text{then} X \sqsubseteq \exists r.(C \sqcap B)$$

[18] TrOWL is a public released reasoner that implements the classification and materialisation mechanism developed in this paper.

In fact, a more generalised variation of the above rule is used in consequence-based algorithms for Horn-$\mathcal{SHIQ}$ [41] and $\mathcal{ALCH}$ [73] to deal with similar forms of interactions between existential restrictions and inverse roles (note that an $\mathcal{ALCH}$ axiom of form $X \sqsubseteq \forall r.B$ is equivalent to the axiom (100) in our example). Nevertheless, the computational price is that such a rule will generate consequences of form $X \sqsubseteq \exists r.(A_1 \sqcap \cdots \sqcap A_n)$ with $n \geq 1$. This results in an exponential increase of the size of the closure. This contributes to the worst-case exponential complexity of the said Horn-$\mathcal{SHIQ}$ and $\mathcal{ALCH}$ algorithms. In order to ensure the tractability of our approach, we did not include such a rule in our rule set.

8.3. Towards Partial Completeness

As we observed in the previous section, there are several patterns of axioms on which our approach will yield incomplete results. Such observations can be exploited to develop partial completeness-guarantee for our approach. To illustrate the idea, in this section, we present a partial completeness-guarantee for restricted $\mathcal{ELHI}$ ontologies based on the pattern shown in Example 11. A comprehensive investigation of the (partial) completeness of tractable algorithms on intractable DLs is beyond the scope of the current paper and will be left for future work.

The DL $\mathcal{ELHI}$ is an extension of $\mathcal{EL}$ with inverse roles. Without lose of generality, in this section we focus on its TBox classification. We also assume such TBoxes consist of only normal form axioms. In $\mathcal{ELHI}$, normal form TBox axioms can be of the following forms:

$$A \sqsubseteq B \tag{104}$$

$$A_1 \sqcap \cdots \sqcap A_n \sqsubseteq B \tag{105}$$

$$A \sqsubseteq \exists R.B \tag{106}$$

$$\exists R.A \sqsubseteq B \text{ where } A \neq \top \tag{107}$$

$$R \sqsubseteq S \tag{108}$$

where $A_{(i)}, B$ can be $\top$, $\bot$ or named concepts, R and S are named roles or their inverse. Note that an axiom of form $\exists R.\top \sqsubseteq B$ is essentially a range restriction of $inv(R)$ and can always be eliminated without affecting the semantics of the TBox [1]. Hence, we omit such a form of axioms here for the sake of conciseness. An example of such an $\mathcal{ELHI}$ TBox is the ontology $\mathcal{O}_{11}$ in Example 11. Following the intractability result of $\mathcal{ELI}$ [1], we know that the worst case complexity for $\mathcal{ELHI}$ TBox classification is exponential.

As Example 11 shows, the incompleteness of our approach on $\mathcal{ELHI}$ occurs when there are interactions between existential restrictions and inverse roles. Such incompleteness can be avoided if we restrict the interactions. There are many ways to achieve this goal. In this section, we employ the idea of *safe role* developed by Carral et al. [12] and use it to establish a sufficient condition for restricted $\mathcal{ELHI}$ in which our approach yields complete results. Particularly, a role R is safe in an $\mathcal{ELHI}$ TBox $\mathcal{O}$ *iff* one of the following two situations occurs:

1. R does not occur in any axiom of form $A \sqsubseteq \exists R.B \in \mathcal{O}$;
2. If there is an axiom $\exists S.A \sqsubseteq B \in \mathcal{O}$, then $R \not\preccurlyeq^* inv(S)$, where $R \preccurlyeq S$ if one of the following situations occurs:
   (a) $R = S$;
   (b) $Inv(R) \sqsubseteq Inv(S)$;
   (c) $R \sqsubseteq S \in \mathcal{O}$;

With this notion, the following theorem shows that our approach is complete for classification of $\mathcal{ELHI}$ TBoxes with only safe roles:

Theorem 17 *Let $\mathcal{O}$ be an $\mathcal{ELHI}$ TBox with only safe roles and $TA_{fa}(\mathcal{O})$ its TBox transformation with name assignment $fa(.)$, then for any $A, B \in \mathcal{CN}_{\mathcal{O}}$, $\mathcal{O} \models A \sqsubseteq B$* iff *$A \sqsubseteq B \in ST(TA_{fa}(\mathcal{O}))$ or $A \sqsubseteq \bot \in ST(TA_{fa}(\mathcal{O}))$.*

The $\leftarrow$ direction of this theorem immediately follows from the soundness of our approach (Theorem 5) therefore we will only need to prove the completeness. The $\rightarrow$ direction of this theorem can be proved with model construction and contrapositive. Particularly, assuming there are $X, Y \in \mathcal{CN}_{\mathcal{O}}$ s.t. $\mathcal{O} \models X \sqsubseteq Y$, $X \sqsubseteq Y \notin ST(TA_{fa}(\mathcal{O}))$ and $X \sqsubseteq \bot \notin ST(TA_{fa}(\mathcal{O}))$, we can show that a model $\mathcal{I}$ of $\mathcal{O}$ can be constructed such that $\mathcal{I} \not\models X \sqsubseteq Y$.

This theorem has the following implications:

1. Given the tractability of our approach, this theorem also confirms part of the results of Carral et al. [12], i.e. $\mathcal{ELHI}$ ontologies with only safe roles can be reasoned with by tractable algorithms with complete results.
2. Note that another tractable description logic DL-Lite$_R$ [10], which is the logic underpinning the OWL 2 QL profile, always satisfies the condition in this theorem. Hence, our approach is also complete for DL-Lite$_R$.
3. The condition identified in this theorem is a sufficient condition but it is not necessary. Our approach may yield complete results even if an ontology does not satisfy the safe-role-only condition. For example, the ontology $\mathcal{O}_{11}$ in Example 11 contains an unsafe role r. It remains unsafe if we extend $\mathcal{O}_{11}$ with an additional axiom $A \sqsubseteq B$. However, with such an extension, our approach will now be able to infer that $X \sqsubseteq D$. This suggests that a weaker completeness guarantee can be developed. We will look into this in our future work.

As we mentioned at the beginning of this section, it is not the focus of this paper to develop a comprehensive completeness guarantee. Therefore, the partial completeness guarantee developed in this section is limited. For example, it does not cover the tractable OWL 2 RL profile. Nevertheless, covering OWL 2 RL with only our approach is not really significant because it can be easily achieved by incorporating into our rule set the complete and tractable completion rules of OWL 2 RL [45]. What will interest us is to identify the conditions in which combinations of $\mathcal{ELHI}$, RL and even more expressive DL features such as cardinality restrictions and complements can be covered by our approach or its extensions. The partial completeness study presented in this section points out a direction to address this problem, i.e. limiting the interactions between different DL features. Particularly, we conjecture that it is possible to extend the result in Theorem 17 to a sufficient condition for completeness guarantee of our approach on DL $\mathcal{SRIQ}$ by eliminating the necessity of performing resolution and cardinality counting. We will look into this problem in our future work.

9. Conclusion & Future Work

In this paper, we presented an approach to addressing the issue of real time approximate reasoning in expressive DL $\mathcal{SROIQ}$, the logic underpinning OWL 2 DL. Although our approach does not produce an optimal approximation like semantic approximation [61], which is expensive to compute, it is soundness preserving, tractable (in PTIME) and produces very high recalls, evidenced by our evaluations on benchmark and real-world ontologies.

The proposed approach consists of two parts. The first part is a transformation from an $\mathcal{SROIQ}$ ontology to an $\mathcal{EL}^{++}$ ontology plus a complement table (CT) maintaining the complementary relations between predicates, a cardinality table (QT) maintaining cardinality information and a inverse table (IT) maintaining inverse relations. Such a transformation can be performed on TBox alone (Def. 4), or on both TBox and ABox (Def. 8). The result is essentially a syntactic variant of the original TBox or ontology that preserves all reasoning results, as we have shown in Theorem 2 and Theorem 8. Therefore this part can be regarded as a normalisation or structural transformation procedure.

The second part of our approach is a set of TBox approximate deduction rules and several more and more optimised ABox approximate deduction procedures with more and more fine-grained completion rules. The purpose of these rule sets is to approximate a fully-fledged $\mathcal{SROIQ}$ materialisation procedure by considering only the interactions between axiom patterns that can be addressed in polynomial time. We show that the optimised ABox reasoning procedures can improve efficiency without losing results compared to the internalisation procedure (Theorem 12, Theorem 14 and Theorem 15) when certain syntactic requirements are met. Particularly, we extend the notion of $\mathcal{EL}^{++}$ nominal-safety (as described in Def. 6, Def. 7 and Theorem 6) to facilitate the optimisation of combined TBox and ABox materialisation with nominals.

Our approach is different from many other approximation approaches in a sense that instead of weakening an ontology in a more expressive language L_1 into a less expressive language L_2 so that it can be reasoned with by a fully-fledged reasoner of L_2, we are actually weakening the reasoning procedure of L_1, granted that this weaker procedure is obtained by extending the reasoning procedure of L_2. To facilitate such an approximate deduction procedure between those of L_1 and L_2, we transform the ontology into L_2-like forms. Nevertheless, such a transformation is purely syntactic and the approximate deduction are actually performed on L_1 axioms. In fact, it is possible to rewrite the rules so that they can be directly used to materialise $\mathcal{SROIQ}$ ontologies.

An important consequence of the above characteristic that distinguishes our approach from the other tractable reasoning solutions is that the tractability is not achieved by restricting the syntax of the ontology, but rather the possible interactions between axioms. Therefore it allows users to construct ontologies with more expressive constructors as they see appropriate but still enjoy the tractable reasoning service with high quality, especially when the expressive constructors in these ontologies only interact with each other in reasoning in limited manners. From our evaluation it can be seen that many real-world ontologies, despite being in very complex DLs, are actually created in this way hence our approach can offer highly efficient reasoning with very high recalls. By further restricting the interactions between different DL features to avoid intractability, it is even possible to devise partial completeness guarantee for our approach (Theorem 17).

Recall that in the introduction we discussed three categories of approaches to ontology reasoning, namely fully-fledged algorithms for expressive DLs, dedicated algorithms for light-weight DLs and approximation-based approaches. Our work belongs to the third category. In the future we would like to further explore its implication on the other two categories. Particularly:

1. Our work approximates a fully-fledged reasoning procedure of $\mathcal{SROIQ}$ with a completion-based algorithm. It is interesting to extend on the current rule set to develop an actual fully-fledged completion-based algorithm for $\mathcal{SROIQ}$ by introducing certain intractable rules, e.g. the ones examined in Sec. 8.2.

2. Following the result of Theorem 17, another interesting research topic is to establish a maximal completeness guarantee for our approach. In other words, we would like to identify a set of conditions such that when an ontology satisfies these conditions, our approach is guaranteed to provide complete materialisation results. The discovery of such conditions essentially means the development of a new tractable DL.

Despite their relevance to the work presented in the current paper, they are exploring either an intractable algorithm, or a new stand-alone tractable DL. Hence they are not within the scope of the current paper and should be investigated as their own tracks of work.

Last but not least, we would like to adopt the idea of parallel reasoning presented by Kazakov et. al. [42] in our approach so that efficiency can be further improved. In this work, the new research challenge is to formulate the rules in ways that parallelisation can be achieved.

Acknowledgements

This work has been partially supported by the European project Marrying Ontologies and Software Technologies (EU ICT 2008-216691), the European project Knowledge Driven Data Exploitation (EU FP7/IAPP 2011-286348), the UK EPSRC project WhatIf (EP/J014354/1).

The authors thank Prof. Ian Horrocks and Dr. Giorgos Stoilos for their helpful discussion on role subsumptions. The authors thank Rafael S. Gonçalves et. al. for providing their hotspots ontologies. The authors also thank BoC-group for providing their ADOxx Metamodelling ontologies.

References

[1] F. Baader, S. Brandt, C. Lutz, Pushing the $\mathcal{EL}$ Envelope, in: Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence IJCAI-05, Morgan-Kaufmann Publishers, Edinburgh, UK, 2005.

[2] F. Baader, S. Brandt, C. Lutz, Pushing the EL Envelope Further, in: K. Clark, P.F. Patel-Schneider (Eds.), In Proceedings of the OWLED 2008 DC Workshop on OWL: Experiences and Directions.

[3] F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi, P.F. Patel-Schneider (Eds.), The Description Logic Handbook: Theory, Implementation, and Applications, Cambridge University Press, 2003.

[4] F. Baader, C. Lutz, B. Suntisrivaraporn, Is Tractable Reasoning in Extensions of the Description Logic $\mathcal{EL}$ Useful in Practice?, in: Proceedings of the Methods for Modalities Workshop (M4M-05), Berlin, Germany.

[5] M. Boddy, T. Dean, Solving time-dependent planning problems, in: Proceedings of the 11th International Joint Conference on Artificial Intelligence - Volume 2, IJCAI'89, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1989, pp. 979–984.

[6] E. Botoeva, D. Calvanese, M. Rodriguez-Muro, Expressive Approximations in DL-Lite Ontologies, in: Proc. of the 14th Int. Conf. on Artificial Intelligence: Methodology, Systems, Applications (AIMSA 2010), volume 6304 of *Lecture Notes in Computer Science*, Springer, 2010, pp. 21–31.

[7] M. Cadoli, F.M. Donini, A survey on knowledge compilation, AI Communications 10 (1997) 137–150.

[8] M. Cadoli, M. Schaerf, Approximation in concept description languages., in: B. Nebel, C. Rich, W.R. Swartout (Eds.), Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning (KR92), Morgan Kaufmann, 1992, pp. 330–341.

[9] M. Cadoli, M. Schaerf, On the complexity of entailment in propositional multivalued logics, Annals of Mathematics and Artificial Intelligence 18 (1996) 29–50.

[10] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, R. Rosati, Tractable reasoning and efficient query answering in description logics: The dl-lite family, Journal of Automated reasoning 39 (2007) 385–429.

[11] D. Calvanese, G.D. Giacomo, D. Lembo, M. Lenzerini, R. Rosati, Data Complexity of Query Answering in Description Logics, Artificial Intelligence 195 (2013) 335 – 360.

[12] D. Carral, C. Feier, B.C. Grau, P. Hitzler, I. Horrocks, EL-ifying Ontologies, in: Automated Reasoning - 7th International Joint Conference, IJCAR 2014.

[13] M. Console, J. Mora, R. Rosati, V. Santarelli, D.F. Savo, Effective computation of maximal sound approximations of description logic ontologies, in: Proc. of the 13th International Semantic Web Conference (ISWC2014), pp. 164–179.

[14] M. Dalal, Anytime families of tractable propositional reasoners, in: Proceedings of the Fourth International Symposium on Artificial Intelligence and Mathematics (AI/MATH-96, pp. 42–45.

[15] M. Dalal, Semantics of an anytime family of reasoners, in: Proceedings of the European Conference on Artificial Intelligence (ECAI'96), pp. 360–364.

[16] J. Dolby, A. Fokoue, A. Kalyanpur, A. Kershenbaum, E. Schonberg, K. Srinivas, L. Ma, Scalable Semantic Retrieval Through Summarization and Refinement, in: Proceedings of the 22nd national conference on Artificial intelligence - Volume 1, AAAI'07, AAAI Press, 2007, pp. 299–304.

[17] F.M. Donini, F. Massacci, EXPTIME Tableaux for ALC, Artif. Intell. 124 (2000) 87–138.

[18] J. Du, G. Qi, J.Z. Pan, Y.D. Shen, A Decomposition-Based Approach to Optimizing Conjunctive Query Answering in OWL DL, in: A. Bernstein, D.R. Karger, T. Heath, L. Feigenbaum, D. Maynard, E. Motta, K. Thirunarayan (Eds.), International Semantic Web Conference, volume 5823 of *Lecture Notes in Computer Science*, Springer, 2009, pp. 146–162.

[19] M. Finger, Towards polynomial approximations of full propositional logic, in: Proceedings of the 17th Brazilian Symposium on Artificial Intelligence, pp. 11–20.

[20] M. Finger, R. Wasserman, Approximate and limited reasoning: Semantics, proof theory, expressivity and control, Journal of Logic And Computation 14 (2004) 179–204.

[21] M. Finger, R. Wassermann, Logics for approximate reasoning: Approximating classical logic "from above", in: Advances in Artificial Intelligence, Springer, 2002, pp. 21–30.

[22] A.M. Frisch, Using model theory to specify ai programs, in: IJCAI, pp. 148–154.

[23] A.M. Frisch, Inference without chaining, in: Proceedings of the 10th International Joint Conference on Artificial Intelligence - Volume 1, IJCAI'87, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987, pp. 515–519.

[24] R.S. Gonçalves, B. Parsia, U. Sattler, Performance Heterogeneity and Approximate Reasoning in Description Logic Ontologies, in: Proceedings of the 11th International Semantic Web Conference (ISWC2012).

[25] P. Groot, H. Stuckenschmidt, H. Wache, Approximating Description Logic Classification For Semantic Web Reasoning, in: Proceedings of the Second European conference on The Semantic Web: research and Applications, ESWC'05, Springer-Verlag, Berlin, Heidelberg, 2005, pp. 318–332.

[26] B.N. Grosof, I. Horrocks, R. Volz, S. Decker, Description Logic Programs: Combining Logic Programs with Description Logic, in: Proceedings of the 12th international conference on World Wide Web, WWW '03, ACM, New York, NY, USA, 2003, pp. 48–57.

[27] HermiT-Benchmark, HermiT Benchmark, `http://hermit-reasoner.com/2009/JAIR_benchmarks/`, 2009.

[28] P. Hitzler, D. Vrandecic, Resolution-Based Approximate Reasoning for OWL DL, in: Y.G. et al. (Ed.), Proceedings of the 4th International Semantic Web Conference, Galway, Ireland, November 2005, volume 3729 of *Lecture Notes in Computer Science*, Springer, Berlin, 2005, pp. 383–397.

[29] B. Hollunder, W. Nutt, M. Schmidt-Schauß, Subsumption Algorithms for Concept Description Languages, in: ECAI, pp. 348–353.

[30] I. Horrocks, O. Kutz, U. Sattler, The Even More Irresistible $\mathcal{SROIQ}$, in: Proc. of the 10th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR 2006), AAAI Press, 2006, pp. 57–67.

[31] I. Horrocks, U. Sattler, Decidability of $\mathcal{SHIQ}$ with Complex Role Inclusion Axioms, Artificial Intelligence 160 (2004) 79–104.

[32] I. Horrocks, U. Sattler, S. Tobies, Practical Reasoning for Very Expressive Description Logics, Logic Journal of the IGPL 8 (2000) 239–264.

[33] I. Horrocks, U. Sattler, S. Tobies, Reasoning with Individuals for the Description Logic $\mathcal{SHIQ}$, in: D. McAllester (Ed.), Proc. of the 17th Int. Conf. on Automated Deduction (CADE 2000), volume 1831 of *Lecture Notes in Computer Science*, Springer, 2000, pp. 482–496.

[34] E.J. Horvitz, Reasoning about beliefs and actions under computational resource constraints, in: In Proceedings of the 1987 Workshop on Uncertainty in Artificial Intelligence, pp. 429–444.

[35] E.J. Horvitz, Reasoning under varying and uncertain resource constraints., in: AAAI, volume 88, pp. 111–116.

[36] A.K. Hudek, G. Weddell, Binary Absorption in Tableaux-based Reasoning for Description Logics, in: Proc. of the 2006 Int. Workshop on Description Logics (DL 2006), volume 189.

[37] U. Hustadt, B. Motik, U. Sattler, Reducing $\mathcal{SHIQ}^-$ Description Logic to Disjunctive Datalog Programs, in: D. Dubois, C.A. Welty, M.A. Williams (Eds.), Proc. of the 9th Int. Conference on Principles of Knowledge Representation and Reasoning (KR 2004), AAAI Press, Whistler, Canada, 2004, pp. 152–162.

[38] H. Kautz, B. Selman, A general framework for knowledge compilation, in: Processing Declarative Knowledge, Springer, 1991, pp. 287–300.

[39] H. Kautz, B. Selman, Forming concepts for fast inference, in: Foundations of Knowledge Representation and Reasoning, Springer, 1994, pp. 200–215.

[40] Y. Kazakov, RIQ and SROIQ are Harder than SHOIQ, in: G. Brewka, J. Lang (Eds.), KR 2008, AAAI Press, 2008, pp. 274–284.

[41] Y. Kazakov, Consequence-Driven Reasoning for Horn SHIQ Ontologies, in: Proceedings of the 21st International Conference on Artificial Intelligence (IJCAI 2009), pp. 2040–2045.

[42] Y. Kazakov, M. Krötzsch, F. Simančík, Concurrent Classification of $\mathcal{EL}$ Ontologies, in: L. Aroyo, C. Welty, H. Alani, J. Taylor, A. Bernstein, L. Kagal, N. Noy, E. Blomqvist (Eds.), Proceedings of the 10th International Semantic Web Conference (ISWC'11), volume 7032 of *LNCS*, Springer, 2011.

[43] Y. Kazakov, M. Krötzsch, F. Simančík, Practical Reasoning with Nominals in the $\mathcal{EL}$ Family of Description Logics, in: G. Brewka, T. Eiter, S.A. McIlraith (Eds.), Proceedings of the 13th International Conference on Principles of Knowledge Representation and Reasoning (KR'12), AAAI Press, 2012, pp. 264–274.

[44] M. Krötzsch, Efficient Rule-Based Inferencing for OWL EL, in: T. Walsh (Ed.), Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI'11), AAAI Press/IJCAI, 2011. 2668–2673.

[45] M. Krötzsch, The not-so-easy task of computing class subsumptions in owl rl, in: The Semantic Web–ISWC 2012, Springer, 2012, pp. 279–294.

[46] M. Krötzsch, S. Rudolph, P. Hitzler, Complexity Boundaries for Horn Description Logics, in: Proceedings of the 22nd AAAI Conference on Artificial Intelligence (AAAI'07), AAAI Press, 2007, pp. 452–457.

[47] H.J. Levesque, A logic of implicit and explicit belief, in: AAAI, pp. 198–202.

[48] H.J. Levesque, Logic and the complexity of reasoning, Journal of Philosophical Logic 17 (1988) 355–389.

[49] H.J. Levesque, A knowledge-level account of abduction., in: IJCAI, pp. 1061–1067.

[50] H.J. Levesque, R.J. Brachman, A fundamental tradeoff in knowledge representation and reasoning, Laboratory for Artificial Intelligence Research, Fairchild, Schlumberger, 1984.

[51] G. LukÁcsy, P. Szeredi, Efficient Description Logic Reasoning in Prolog: The DLog System, Theory Pract. Log. Program. 9 (2009) 343–414.

[52] C. Lutz, D. Walther, F. Wolter, Conservative Extensions in Expressive Description Logics, in: M. Veloso (Ed.), Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI'07), AAAI Press, 2007, pp. 453–458.

[53] F. Massacci, Efficient Approximate Deduction and an Application to Computer Security, Ph.D. thesis, Universitá di Roma I "La Sapienza", Dipartimento di Informatica e Sistemistica, 1998.

[54] J.A. Miñarro-Giménez, K. Blagec, R.D. Boyce, K.P. Adlassnig, M. Samwald, An ontology-based, mobile-optimized system for pharmacogenomic decision support at the point-of-care, PloS one 9 (2014) e93769.

[55] B. Motik, Reasoning in Description Logics using Resolution and Deductive Databases, Ph.D. thesis, Univesität Karlsruhe (TH), Karlsruhe, Germany, 2006.

[56] B. Motik, B.C. Grau, I. Horrocks, Z. Wu, A. Fokoue, C. Lutz, OWL 2 Web Ontology Language: Profiles (Second Edition), W3C Recommendation, W3C, 2012.

[57] B. Motik, P.F. Patel-Schneider, B. Parsia, C. Bock, A. Fokoue, P. Haase, R. Hoekstra, I. Horrocks, A. Ruttenberg, U. Sattler, M. Smith, OWL 2 Web Ontology Language: Structural Specification and Functional-Style Syntax (Second Edition), W3C Recommendation, W3C, 2012.

[58] B. Motik, U. Sattler, A Comparison of Reasoning Techniques for Querying Large Description Logic ABoxes, in: M. Hermann, A. Voronkov (Eds.), Proc. of the 13th Int. Conference on Logic for Programming Artificial Intelligence and Reasoning (LPAR 2006), volume 4246 of *LNCS*, Springer, Phnom Penh, Cambodia, 2006, pp. 227–241.

[59] B. Motik, R. Shearer, I. Horrocks, Hypertableau Reasoning for Description Logics, Journal of Artificial Intelligence Research 36 (2009) 165–228.

[60] J.Z. Pan, Y. Ren, N. Jekjantuk, J. Garcia, Reasoning the fma ontologies with trowl., in: Proceedings of OWL Reasoner Evaluation Workshop, pp. 107–113.

[61] J.Z. Pan, E. Thomas, Approximating OWL-DL Ontologies, in: Proceedings of the 22nd National Conference on Artificial Intelligence - Volume 2, AAAI'07, AAAI Press, 2007, pp. 1434–1439.

[62] J.Z. Pan, E. Thomas, Y. Ren, S. Taylor., Tractable Fuzzy and Crisp Reasoning in Ontology Applications, in: IEEE Computational Intelligence Magazine.

[63] F.S. Parreiras, S. Staab, Using Ontologies With UML Class-based Modeling: The TwoUse Approach, Data & Knowledge Engineering 69 (2010) 1194 – 1207. Special issue on contribution of ontologies in designing advanced information systems.

[64] P.F. Patel-Schneider, A four-valued semantics for terminological logics, Artificial Intelligence 38 (1989) 319–351.

[65] Y. Ren, J.Z. Pan, Y. Zhao, Soundness Preserving Approximation for TBox Reasoning, in: Proceedings of the 24th AAAI Conference on Artificial Intelligence (AAAI'10).

[66] Y. Ren, J.Z. Pan, Y. Zhao, Towards Soundness Preserving Approximation for ABox Reasoning of OWL2, in: Proceedings of Description Logics Workshop 2010 (DL2010).

[67] A.A. Romero, B.C. Grau, I. Horrocks, MORe: Modular Combination of OWL Reasoners for Ontology Classification, in: Proceedings of the 11th International Semantic Web Conference (ISWC 2012), LNCS, Springer, 2012.

[68] S. Rudolph, T. Tserendorj, P. Hitzler, What Is Approximate Reasoning?, in: D. Calvanese, G. Lausen (Eds.), Proceedings of the 2nd International Conference on Web Reasoning and Rule Systems (RR2008), volume 5341 of *LNCS*, Springer, 2008, pp. 150–164.

[69] A. Schaerf, Reasoning With Individuals in Concept Languages, Data and Knowledge Engineering 13 (1994) 141–176.

[70] M. Schaerf, M. Cadoli, Tractable reasoning via approximation, Artificial Intelligence 74 (1995) 249 – 310.

[71] B. Selman, H. Kautz, Knowledge compilation using horn approximations, in: Proceedings of the Ninth National Conference on Artificial Intelligence - Volume 2, AAAI'91, AAAI Press, 1991, pp. 904–909.

[72] B. Selman, H. Kautz, Knowledge Compilation and Theory Approximation, Journal of the ACM (JACM) 43 (1996) 193–224.

[73] F. Simančík, Y. Kazakov, I. Horrocks, Consequence-Based Reasoning beyond Horn Ontologies, in: Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume Two, pp. 1093–1099.

[74] E. Sirin, B. Cuenca Grau, B. Parsia, From Wine to Water: Optimizing Description Logic Reasoning for Nominals, in: Proceedings of KR-2006, Tenth International Conference on Principles of Knowledge Representation and Reasoning, Lake District of the United Kingdom, June 2-5, 2006, AAAI Press, 2006, pp. 90–99.

[75] M.Q. Stearns, C. Price, K.A. Spackman, A.Y. Wang, SNOMED Clinical Terms: Overview of the Development Process and project Status, in: Proceedings of the AMIA Symposium, American Medical Informatics Association, p. 662.

[76] H. Stuckenschmidt, F. van Harmelen, Approximating Terminological Queries, in: H.L. et al. (Ed.), Proceedings of the Proceedings of the 4th International Conference on Flexible Query Answering Systems (FQAS'02), number 2522 in Advances in Soft Computing, Springer-Verlag, 2002, pp. 329–343.

[77] B. Suntisrivaraporn, Module Extraction and Incremental Classification: A Pragmatic Approach for $\mathcal{EL}^+$ Ontologies, in: S. Bechhofer, M. Hauswirth, J. Hoffmann, M. Koubarakis (Eds.), Proceedings of the 5th European Semantic Web Conference (ESWC'08), volume 5021 of *Lecture Notes in Computer Science*, Springer-Verlag, 2008, pp. 230–244.

[78] E. Thomas, J.Z. Pan, Y. Ren, TrOWL: Tractable OWL 2 Reasoning Infrastructure, in: the Proc. of the Extended Semantic Web Conference (ESWC2010).

[79] D. Tsarkov, I. Horrocks, Efficient Reasoning with Range and Domain Constraints, in: Proc. of the 2004 Description Logic Workshop (DL 2004), pp. 41–50.

[80] D. Tsarkov, I. Horrocks, P.F. Patel-Schneider, Optimizing Terminological Reasoning for Expressive Description Logics, J. of Automated Reasoning 39 (2007) 277–316.

[81] T. Tserendorj, S. Rudolph, M. Krötzsch, P. Hitzler, Approximate OWL-Reasoning with Screech, in: D. Calvanese, G. Lausen (Eds.), Web Reasoning and Rule Systems, Second International Conference, RR 2008, Karlsruhe, Germany, October 31-November 1, 2008. Proceedings, volume 5341 of *LNCS*, Springer, 2008, pp. 165–180.

[82] A. Urquhart, Hard examples for resolution, Journal of the ACM (JACM) 34 (1987) 209–219.

[83] H. Wache, P. Groot, H. Stuckenschmidt, Scalable Instance Retrieval for the Semantic Web by Approximation, in: Proceedings of the 2005 international conference on Web Information Systems Engineering, WISE'05, Springer-Verlag, Berlin, Heidelberg, 2005, pp. 245–254.

[84] T. Walter, F. Silva Parreiras, S. Staab, OntoDSL: An Ontology-Based Framework for Domain-Specific Languages, in: Proceedings of the 12th International Conference on Model Driven Engineering Languages and Systems, MODELS '09, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 408–422.

Appendix

In this appendix we provide proofs for all propositions, lemmas and theorems in the paper.

The proofs for Proposition 1, Proposition 3, Theorem 4, Theorem 5, Proposition 7, Proposition 9, Theorem 10 and Theorem 11 are relative simple. The other proofs are more sophisticated.

Proposition 1 ***(TBox Linear Transformation)*** *In Def. 4, transforming from $\mathcal{O}$ to $TA_{fa}(\mathcal{O})$ can be done in linear time.*

Proof We prove the proposition by showing that for any ontology $\mathcal{O}$ and its TBox transformation $(\mathcal{T}, CT, QT, IT)$, if $\mathcal{O}$ contains $n_{\mathcal{O}}$ number of terms, then we have:

1. $|\mathcal{T}| \leq n_{\mathcal{O}} + |\mathcal{O}|$;
2. $|CT| = n_{\mathcal{O}}$;
3. $|QT| \leq n_{\mathcal{O}}$;
4. $|IT| = 4 \times |RN_{\mathcal{O}}|$.

where $|\mathcal{T}|$ and $|\mathcal{O}|$ are the numbers of axioms in $\mathcal{T}$ and $\mathcal{O}$, respectively. $|CT|$ ($|QT|$, $|IT|$, $|\mathcal{RN}_{\mathcal{O}}|$) is the number of pairs (triples, or elements) in CT (QT, IT, $\mathcal{RN}_{\mathcal{O}}$).

Particularly:

1. $|\mathcal{T}|$: For each axiom in $\mathcal{O}$, at most one axiom is added into $\mathcal{T}$ (Step-2 to 4). For each term in $\mathcal{O}$, at most one axiom is added into $\mathcal{T}$ (Step-5). Together $|\mathcal{T}| \leq n_{\mathcal{O}} + |\mathcal{T}_{\mathcal{O}}|$.
2. $|CT|$: For each term in $\mathcal{O}$, one element is added into CT (Step-5).
3. $|QT|$: For each term in $\mathcal{O}$, at most one element is added into QT (Step-5.c.iii).
4. $|IT|$: For each role in $\mathcal{O}$, four elements are added into IT (Step-6). □

Theorem 2 ***(TBox Reasoning Preservation)*** *For any ontology $\mathcal{O} = (\mathcal{T}_{\mathcal{O}}, \mathcal{A}_{\mathcal{O}})$ and its TBox transformation $(\mathcal{T}, CT, QT, IT)$, let $\mathcal{T}'$ be a TBox constructed as follows:*

1. *$\mathcal{T}'$ is initialised as $\mathcal{T}$;*
2. *for any concept pair $(A, B) \in CT$, $\mathcal{T}' = \mathcal{T}' \cup \{A \equiv \neg B\}$;*
3. *for any role pair $(r, s) \in CT$, $\mathcal{T}' = \mathcal{T}' \cup \{Dis(r, s)\}$;*
4. *for any $(A, B, r, n) \in QT$, $\mathcal{T}' = \mathcal{T}' \cup \{A \equiv\geq nr.B\}$;*
5. *for any $(r, s) \in IT$, $\mathcal{T}' = \mathcal{T}' \cup \{r \equiv Inv(s)\}$;*

then for any GCI α with $\Sigma_\alpha \subseteq \Sigma_{\mathcal{T}_{\mathcal{O}}}$, we have $\mathcal{T}_{\mathcal{O}} \models \alpha$ iff $\mathcal{T}' \models \alpha$:

Proof This theorem can be proved by showing that $\mathcal{T}_{\mathcal{O}} \cup \mathcal{T}' \cup \mathcal{T}_{fa}$ is a $\Sigma_{\mathcal{T}_{\mathcal{O}}}$-conservative extension [52] of both $\mathcal{T}_{\mathcal{O}}$ and $\mathcal{T}'$, where $\mathcal{T}_{fa}$ is the set of definition $fa(P) \equiv P$ for all term P of $\mathcal{O}$ except complement of roles. A TBox $\mathcal{T}_1$ is a Σ-conservative extension of another TBox $\mathcal{T}_2$ for $\Sigma \subseteq \Sigma_{\mathcal{T}_2}$ iff $\mathcal{T}_2 \subseteq \mathcal{T}_1$ and for any GCI α with $\Sigma_\alpha \in \Sigma$, $\mathcal{T}_1 \models \alpha$ implies $\mathcal{T}_2 \models \alpha$.

We first show that $\mathcal{T}_{\mathcal{O}} \cup \mathcal{T}_{fa} \cup \mathcal{T}'$ is a conservative extension (CE) of $\mathcal{T}_{\mathcal{O}}$ by induction. Apparently $\mathcal{T}_{\mathcal{O}} \subseteq \mathcal{T}_{\mathcal{O}} \cup \mathcal{T}' \cup \mathcal{T}_{fa}$ and $\mathcal{T}_{\mathcal{O}} \cup \mathcal{T}_{fa}$ is a CE of $\mathcal{T}_{\mathcal{O}}$ since it only adds definitions of fresh names. Assuming some $\mathcal{T}_i$ with $\mathcal{T}_{\mathcal{O}} \cup \mathcal{T}_{fa} \subseteq \mathcal{T}_i \subseteq \mathcal{T}_{\mathcal{O}} \cup \mathcal{T}_{fa} \cup \mathcal{T}'$ is a CE of $\mathcal{T}_{\mathcal{O}}$, an axiom $\beta \in \mathcal{T}' \setminus \mathcal{T}_i$ can be of the following origins:

1. β is generated by Step-2 or Step-3 of Def. 4: β simply rewrites an axiom in $\mathcal{T}_{\mathcal{O}}$ with the name assignments of the corresponding LHS and RHS of the axiom. Given that all name assignments are defined by $\mathcal{T}_{fa}$ except those of complement of roles, and complement of roles don't appear in $\mathcal{T}_{\mathcal{O}}$, we have $\mathcal{T}_i \models \beta$. Hence $\mathcal{T}_i \cup \{\beta\}$ is still a CE of $\mathcal{T}_{\mathcal{O}}$;

2. β is generated by Step-4 of Def. 4: β is of form $r \sqsubseteq s$ where r is a named assignment defined by $\mathcal{T}_{fa}$ and s as the name assignment of some role S do not appear anywhere in $\mathcal{T}_{\mathcal{O}}$, we shall still have $\mathcal{T}_i \cup \{\beta\}$ a CE of $\mathcal{T}_{\mathcal{O}}$;
3. β is generated by Step-5.(a) of Def. 4: Given that $\mathcal{T}_{fa} \subseteq \mathcal{T}_i$ we know that $\mathcal{T}_i \models \beta$ hence $\mathcal{T}_i \cup \{\beta\}$ is still a CE of $\mathcal{T}_{\mathcal{O}}$;
4. β is generated by Step-5.(b) of Def. 4: here we have two possibilities:
   (a) β is the axiom $fa(C) \equiv \exists fa(R).fa(D)$: since $C \equiv \exists R.D$ is a tautology and $\mathcal{T}_{fa} \subseteq \mathcal{T}_i$, we have $\mathcal{T}_i \models \beta$ hence $\mathcal{T}_i \cup \{\beta\}$ is a CE of $\mathcal{T}_{\mathcal{O}}$;
   (b) β is the axiom $\exists fa(Inv(R)).fa({\sim}C) \sqsubseteq fa({\sim}D)$: from $C \equiv \exists R.D$ we have $\mathcal{T}_i \models \exists R.D \sqsubseteq C$, hence $\mathcal{T}_i \models {\sim}C \sqsubseteq \forall R.{\sim}D$. This further implies $\mathcal{T}_i \models \exists Inv(R).{\sim}C \sqsubseteq {\sim}D$. Together with $\mathcal{T}_{fa} \subseteq \mathcal{T}_i$ we have $\mathcal{T}_i \models \beta$ hence $\mathcal{T}_i \cup \{\beta\}$ is a CE of $\mathcal{T}_{\mathcal{O}}$;
5. β is generated by Step-5.(c).i of Def. 4, then β is a tautology.
6. β is generated by Step-5.(c).ii of Def. 4, then it is similar to the above situation of 5.(b) that $\mathcal{T}_i \cup \{\beta\}$ is still a CE of $\mathcal{T}_{\mathcal{O}}$;
7. β is generated by Step-5.(c).iii of Def. 4: since $C \equiv \geq nR.D$ is a tautology, we have $\mathcal{T}_i \models C \sqsubseteq \exists R.D$. With $\mathcal{T}_{fa} \subseteq \mathcal{T}_i$ it is easy to show that $\mathcal{T}_i \models \beta$ hence $\mathcal{T}_i \cup \{\beta\}$ is still a CE of $\mathcal{T}_{\mathcal{O}}$;
8. β is generated by Step-2 in Theorem 2: apparently there is a concept term C of $\mathcal{O}$ s.t. $\beta = fa(C) \sqsubseteq \neg fa({\sim}C)$. Given that $\mathcal{T}_{fa} \subseteq \mathcal{T}_i$, we have $\mathcal{T}_i \models \beta$ and hence $\mathcal{T}_i \cup \{\beta\}$ is still a CE of $\mathcal{T}_{\mathcal{O}}$;
9. β is generated by Step-3 in Theorem 2: apparently there is a role term R of $\mathcal{O}$ s.t. $\beta = Dis(fa(R), fa({\sim}R))$. Since complement of roles are not supported in $\mathcal{SROIQ}$ TBox, either $fa(R)$ or $fa({\sim}R)$ is a fresh role name that do not appear in $\mathcal{T}_{\mathcal{O}}$. Hence $\mathcal{T}_{fa} \cup \{\beta\}$ is still a CE of $\mathcal{T}_{\mathcal{O}}$;
10. β is generated by Step-4 in Theorem 2: apparently there is a concept term $C = \geq nR.D$ in $\mathcal{O}$ s.t. $\beta = fa(C) \equiv \geq nfa(R).fa(D)$. Given that $\mathcal{T}_{fa} \subseteq \mathcal{T}_i$, we have $\mathcal{T} \models \beta$ and hence $\mathcal{T}_i \cup \{\beta\}$ is still a CE of $\mathcal{T}_{\mathcal{O}}$;
11. β is generated by Step-5 in Theorem 2: apparently there is a role term R in $\mathcal{O}$ s.t. $\beta = fa(R) \equiv Inv(()fa(Inv(R)))$. Given that $\{R \equiv fa(R), fa(Inv(R)) \equiv riR\} \subseteq \mathcal{T}_{fa} \subseteq \mathcal{T}_i$, we have $\mathcal{T}_i \models \beta$ and hence $\mathcal{T}_i \cup \{\beta\}$ is still a CE of $\mathcal{T}_{\mathcal{O}}$;

Together we can show that $\mathcal{T}_{\mathcal{O}} \cup \mathcal{T}_{fa} \cup \mathcal{T}'$ is a CE of $\mathcal{T}_{\mathcal{O}}$.

We can show that $\mathcal{T}_{\mathcal{O}} \cup \mathcal{T}_{fa} \cup \mathcal{T}'$ is a CE of $\mathcal{T}'$ by showing that $\mathcal{T}' \models \mathcal{T}_{fa}$ and $\mathcal{T}' \cup \mathcal{T}_{fa} \models \mathcal{T}_{\mathcal{O}}$.

For every axiom $\beta = fa(P) \equiv P \in \mathcal{T}_{fa}$, we perform structural induction on P. Note that P is in negation normal form hence P may contain constructors such as $\sqcup$, $\forall$, $\leq$:

1. If $P \in \mathcal{CN}_{\mathcal{O}} \cup \mathcal{RN}_{\mathcal{O}}$, according to Def. 3 β is a tautology and obviously $\mathcal{T}' \models \beta$;
2. If P is of form $\neg C$, $\beta = fa(\neg C) \equiv \neg C$. Since C is also a term, by structural induction we have $\mathcal{T}' \models fa(C) \equiv C$. We also have $(C, \neg C) \in CT$, hence by Step-2 of Theorem 2 we have $fa(C) \equiv \neg fa(\neg C) \in \mathcal{T}'$. Together we have $\mathcal{T}' \models \beta$;
3. If P is of form $C \sqcap D$, $\beta = fa(C \sqcap D) \equiv C \sqcap D$. Since C and D are also terms, by structural induction we have $\mathcal{T}' \models fa(C) \equiv C$ and $\mathcal{T}' \models fa(D) \equiv D$. By Step-5.(a) of Def. 4 we have $fa(C \sqcap D) \equiv fa(C) \sqcap fa(D) \in \mathcal{T}'$. Together we have $\mathcal{T}' \models \beta$;
4. If P is of form $C \sqcup D$, $\beta = fa(C \sqcup D) \equiv C \sqcup D$. Since ${\sim}C$ and ${\sim}D$ are terms, by structural induction we have $\mathcal{T}' \models \{fa({\sim}C) \equiv \neg C, fa({\sim}D) \equiv \neg D\}$. Since ${\sim}C \sqcap {\sim}D$ is also a term, by Def. 4 we also have $fa({\sim}C \sqcap {\sim}D) \equiv fa({\sim}C) \sqcap fa({\sim}D)$. By Step-2 of Theorem 2 we also have $fa(C \sqcup D) \equiv \neg fa({\sim}C \sqcap {\sim}D) \in \mathcal{T}'$. Together we can infer that $\mathcal{T}' \models \beta$;

5. If P is of form $\exists R.C$, $\beta = fa(\exists R.C) \equiv \exists R.C$. Both R and C are terms. By structural induction we have $\mathcal{T}' \models fa(R) \equiv R$ and $fa(C) \equiv C$. We also have $fa(\exists R.C) \equiv \exists fa(R).fa(C) \in \mathcal{T}'$. Together we have $\mathcal{T}' \models \beta$;
6. If P is of form $\forall R.C$, $\beta = fa(\forall R.C) \equiv \forall R.C$. R and $\sim C$ are terms hence we have $\mathcal{T}' \models \{fa(R) \equiv R, fa(\sim C) \equiv \neg C\}$. $\exists R.\sim C$ is also a term, by Def. 4 we have $fa(\exists R.\sim C) \equiv \exists fa(R).fa(\sim C) \in \mathcal{T}'$. By Step-2 of Theorem 2 we also have $fa(\forall R.C) \equiv \neg fa(\exists R.\sim C) \in \mathcal{T}'$. Together we have $\mathcal{T}' \models \beta$;
7. If P is of form $\geq nS.C$, $\beta = fa(\geq nS.C) \equiv \geq nS.C$. S and C are terms hence we have $\mathcal{T}' \models \{fa(R) \equiv R, fa(C) \equiv C\}$. By Def. 4 we have $(fa(\geq nS.C), fa(C), fa(S), n) \in QT$. From Step-4 of Theorem 2 we also have $fa(\geq nS.C) \equiv \geq nfa(S).fa(C) \in \mathcal{T}'$. Together we have $\mathcal{T}' \models \beta$;
8. If P is of form $\leq nS.C$, $\beta = fa(\leq nS.C) \equiv \leq nS.C$. S and C are terms hence we have $\mathcal{T}' \models \{fa(R) \equiv R, fa(C) \equiv C\}$. $\geq n+1S.C$ is also a term. By Def. 4 we have $(fa(\geq n+1S.C), fa(C), fa(S), n+1) \in QT$. From Step-4 of Theorem 2 we further have $fa(\geq n+1S.C) \equiv \geq n+1fa(S).fa(C) \in \mathcal{T}'$. From Step-2 of Theorem 2 we also have $fa(\leq nS.C) \equiv \neg fa(\geq n+1S.C)$. Together we have $\mathcal{T}' \models \beta$;
9. If P is of form R^-, $\beta = fa(R^-) \equiv R^-$. R is term hence we have $\mathcal{T}' \models fa(R) \equiv R$. By Def. 4 we have $(fa(R), fa(R^-)) \in IT$. Together with Step-5 of Theorem 2 we have $fa(R^-) \equiv fa(R)^-$. Together we have $\mathcal{T}' \models \beta$.

By induction we can show that $\mathcal{T}' \models \mathcal{T}_{fa}$.

For every axiom $\beta \in \mathcal{T}_{\mathcal{O}}$, β can be either a GCI, or an RI, or a role disjointness axiom:

1. If β is a GCI or an RI, then it is obvious that $\mathcal{T}' \cup \mathcal{T}_{fa} \models \beta$ because $\mathcal{T}'$ contains β with expressions replaced by names and $\mathcal{T}_{fa}$ contains the equivalence between expressions and corresponding names;
2. If β is of form $Dis(R, S)$. From Step-4 of Def. 4 we have $fa(R) \sqsubseteq fa(\neg S) \in \mathcal{T}'$. From Step-3 of Theorem 2 we have $Dis(fa(S), fa(\neg S)) \in \mathcal{T}'$. We also have $fa(R) \equiv R$ and $fa(S) \equiv S \in \mathcal{T}_{fa}$. By semantics of role disjointness we have $\mathcal{T}' \cup \mathcal{T}_{fa} \models \beta$.

Together we have $\mathcal{T}' \cup \mathcal{T}_{fa} \models \mathcal{T}_{\mathcal{O}}$. Since we also have $\mathcal{T}' \models \mathcal{T}_{fa}$ we have $\mathcal{T}' \models \mathcal{T}_{fa} \cup \mathcal{T}_{\mathcal{O}}$. Hence we know that $\mathcal{T}' \cup \mathcal{T}_{fa} \cup \mathcal{T}_{\mathcal{O}}$ is a CE of $\mathcal{T}'$.

Note that $\Sigma_{\mathcal{T}_{\mathcal{O}}} \subseteq \Sigma_{\mathcal{T}'}$. Together with the conclusion that $\mathcal{T}' \cup \mathcal{T}_{fa} \cup \mathcal{T}_{\mathcal{O}}$ is also a CE of $\mathcal{T}_{\mathcal{O}}$, we can infer that for any GCI α with $\Sigma_\alpha \subseteq \Sigma_{\mathcal{T}_{\mathcal{O}}}$, we have $\mathcal{T}_{\mathcal{O}} \models \alpha$ iff $\mathcal{T}' \cup \mathcal{T}_{fa} \cup \mathcal{T}_{\mathcal{O}} \models \alpha$ iff $\mathcal{T}' \models \alpha$. □

Proposition 3 ***(TBox Approximation)*** *For an ontology $\mathcal{O}$, let $TA_{fa}(\mathcal{O})$ be $(\mathcal{T}, CT, QT, IT)$ as defined in Def. 4, then $\mathcal{T}$ is an $\mathcal{EL}^{++}$ TBox such that for any axiom α with $\Sigma_\alpha \subseteq \Sigma_{\mathcal{O}}$, we have $\mathcal{T} \models \alpha$ only if $\mathcal{O} \models \alpha$.*

Proof It is obvious that $\mathcal{T}$ only contains $\mathcal{EL}^{++}$ TBox axioms.

The soundness of $\mathcal{T}$ w.r.t. $\mathcal{O}$ is an apparent corollary of Theorem 2 since $\mathcal{T} \subseteq \mathcal{T}'$, which preserves all semantics of $\mathcal{T}_{\mathcal{O}}$, the TBox of $\mathcal{O}$. □

Theorem 4 ***(TBox Approximate Reasoning Complexity)*** *For an ontology $\mathcal{O}$, its TBox Approximate Reasoning Closure $ST(TA_{fa}(\mathcal{O}))$ can be computed in polynomial time w.r.t. the size of $\mathcal{O}$.*

Proof Assume $TA_{fa} = (\mathcal{T}, CT, QT, IT)$, from Proposition 1 we know that $\mathcal{T}$, CT, QT and IT are all in linear size, and can be generated in linear time w.r.t. the size of $\mathcal{O}$. Also, the normalisation of $\mathcal{T}$ can be performed in linear time.

It is sufficient to prove this theorem by showing that $ST(TA_{fa}(\mathcal{O}))$ can be computed in polynomial time w.r.t. $|\mathcal{CN}_{\mathcal{T}}| + |\mathcal{RN}_{\mathcal{T}}|$: First of all, it is easy to implement an algorithm that initialises $ST(TA_{fa}(\mathcal{O}))$ as specified by Step-1 in Def. 5. Then the algorithm repeatedly applies all rules until no more changes can be made on $ST(TA_{fa}(\mathcal{O}))$.

Due to the following facts that:

1. there are at most $|\mathcal{CN}_{\mathcal{T}}| \times |\mathcal{CN}_{\mathcal{T}}|$ atomic concept subsumptions of the form $A \sqsubseteq B$;
2. there are at most $|\mathcal{CN}_{\mathcal{T}}| \times |\mathcal{RN}_{\mathcal{T}}| \times |\mathcal{CN}_{\mathcal{T}}|$ concept inclusions of the form $A \sqsubseteq \exists r.B$;
3. there are at most $|\mathcal{RN}_{\mathcal{T}}| \times |\mathcal{RN}_{\mathcal{T}}|$ role inclusions of the form $r \sqsubseteq s$;
4. there are at most $|\mathcal{RN}_{\mathcal{T}}|^3$ role inclusions of the form $r_1 \circ r_2 \sqsubseteq_* r_3$;
5. each rule application generates one axiom of one of the above forms, and once an axiom is inferred it will not be removed.

It is obvious that the application of rules will be performed in polynomial times w.r.t. $|\mathcal{CN}_{\mathcal{T}}| + |\mathcal{RN}_{\mathcal{T}}|$. The theorem can thus be proved. □

Theorem 5 ***(TBox Approximation Reasoning Soundness)*** *For an ontology $\mathcal{O}$, let α be an axiom such that $\Sigma_\alpha \subseteq \Sigma_{\mathcal{O}}$, then $\alpha \in ST(TA_{fa}(\mathcal{O}))$ only if $\mathcal{O} \models \alpha$.*

Proof This theorem directly follows from the facts:

1. As Theorem 2 shows, the transformation in Def.4 is a syntactic variant of the original TBox that replaces each concept expression with a atomic concept, each role expression with an atomic role. Additional axioms are essentially tautologies.
2. The normalisation of the transformed TBox is also a conservative rewriting that does not affect soundness of entailments constructed using $\Sigma_{\mathcal{O}}$.
3. All the **R** rules preserve the correctness of the results. □

Theorem 6 ***(Syntactic Nominal-safety Under Approximate Reasoning)*** *For an ontology $\mathcal{O}$, let $TA_{fa}(\mathcal{O}) = (\mathcal{T}, CT, QT, IT)$, then $\mathcal{O}$ is semantic nominal-safe under approximate reasoning if $\mathcal{T}$ satisfies the following syntactic properties, where A is a concept such that A, $fc(A) \in \mathcal{CN}_{\mathcal{T}}$, $B_{(i)}$ is an arbitrary concept, a and b are individuals and r is a named role:*

1. *$\mathcal{T}$ is a nominal-safe $\mathcal{EL}^{++}$ TBox;*
2. *$fc(\{a\}) \sqsubseteq A \notin \mathcal{T}$;*
3. *$fc(\{a\}) \sqsubseteq \{b\} \notin \mathcal{T}$;*
4. *$B_1 \sqcap \cdots \sqcap fc(\{a\}) \sqcap \cdots \sqcap B_n \sqsubseteq \bot \notin \mathcal{T}$;*
5. *$fc(\{a\}) \sqsubseteq \exists r.A \notin \mathcal{T}$;*
6. *$fc(\{a\}) \sqsubseteq \exists r.\{b\} \notin \mathcal{T}$;*
7. *$fc(\{a\}) \sqsubseteq \exists r.fc(\{b\}) \notin \mathcal{T}$;*

Proof We use TA to denote $TA_{fa}(\mathcal{O}) = (\mathcal{T}, CT, QT, IT)$ and we use ST to denote $ST(TA_{fa}(\mathcal{O}))$. Assuming ST is obtained by applying **R** rules for n times, we use ST_i to denote the state of ST after i application of **R** rules. Obviously, $ST_0 = \mathcal{T}$ and $ST_n = ST$.

This theorem can be proved by showing that when the conditions in theorem are satisfied, the following stronger claims hold:

$$\begin{aligned}
&A \sqsubseteq \{a\} \notin ST_n;\\
&fc(\{a\}) \sqsubseteq A \notin ST_n;\\
&fc(\{a\}) \sqsubseteq \{b\} \notin ST_n;\\
&fc(\{a\}) \sqsubseteq \exists r.A \notin ST_n;\\
&fc(\{a\}) \sqsubseteq \exists r.\{b\} \notin ST_n;\\
&fc(\{a\}) \sqsubseteq \exists r.fc(\{b\}) \notin ST_n;
\end{aligned}$$

We prove the above claims by induction on the number of reasoning steps. First of all, all claims hold in ST_0. Now we assume all claims hold in ST_k, e.g. $A \sqsubseteq \{a\} \notin ST_k$, and investigate the inclusion of these entailments in ST_{k+1} via different rules.

$A \sqsubseteq \{a\}$ can be derived by **R1, 2, 4, 10, 11, 12**:

1. **R1** is not applicable due to the hypothesis on $A \sqsubseteq \{a\}$ and $fc(\{a\}) \sqsubseteq \{a\}$;
2. **R2, 4** are not applicable due to the condition that $\mathcal{T}$ is a nominal-safe $\mathcal{EL}^{++}$ TBox hence GCIs with $\{a\}$ as the right-hand side do not appear in $\mathcal{T}$;
3. **R10** is not applicable due to the hypothesis on $fc(\{a\}) \sqsubseteq A$;
4. **R11** is not applicable due to the condition that $B_1 \sqcap \cdots \sqcap fc(\{a\}) \sqcap \cdots \sqcap B_n \sqsubseteq \bot \notin \mathcal{T}$;
5. **R12** is not applicable due to the hypothesis on $fc(\{a\}) \sqsubseteq \exists r.\{b\}$;

Together we know that $A \sqsubseteq \{a\} \notin ST_{k+1}$.

$fc(\{a\}) \sqsubseteq A$ can be derived by **R1, 2, 4, 10, 11, 12**:

1. **R1** is not applicable due to the hypothesis on $fc(\{a\}) \sqsubseteq A$ and $fc(\{a\}) \sqsubseteq \{b\}$;
2. **R2** is not applicable due to the same reason as above, except the situation that $\bigsqcap_{i=1}^{n} fc(\{b_i\}) \sqsubseteq A \in \mathcal{T}$ and $fc(\{a\}) \sqsubseteq fc(\{b_i\}) \in ST_k$ for $i = 1, \ldots, n$. In this case, we essentially have $a \doteq b_i$ and $fc(\{a\}) \sqsubseteq A \in ST_k$, which is against the hypothesis;
3. **R4** is not applicable due to the hypothesis on $fc(\{a\}) \sqsubseteq \exists r.A$, $fc(\{a\}) \sqsubseteq \exists r.\{b\}$ and $fc(\{a\}) \sqsubseteq \exists r.fc(\{b\})$;
4. **R10** is not applicable due to the hypothesis on $A \sqsubseteq \{a\}$;
5. **R11** is not applicable due to the condition that that $B_1 \sqcap \cdots \sqcap fc(\{a\}) \sqcap \cdots \sqcap B_n \sqsubseteq \bot \notin \mathcal{T}$ and the hypothesis on $fc(\{a\}) \sqsubseteq A$ and $fc(\{a\}) \sqsubseteq \{b\}$;
6. **R12** is not applicable due to the hypothesis on $fc(\{a\}) \sqsubseteq \exists r..\{b\}$;

Together we know that $fc(\{a\}) \sqsubseteq A \notin ST_{k+1}$.

$fc(\{a\}) \sqsubseteq \{b\}$ can be derived by **R1, 2, 4, 10, 11, 12**:

1. **R1** is not applicable due to the hypothesis on $fc(\{a\}) \sqsubseteq A$ and $fc(\{a\}) \sqsubseteq \{b\}$;
2. **R2, 4** are not applicable due to the condition that $\mathcal{T}$ is a nominal-safe $\mathcal{EL}^{++}$ TBox.
3. **R10** is not applicable due to the hypothesis on $fc(\{a\}) \sqsubseteq \{b\}$;
4. **R11** is not applicable due to the condition that $B_1 \sqcap \cdots \sqcap fc(\{a\}) \sqcap \cdots \sqcap B_n \sqsubseteq \bot \notin \mathcal{T}$;
5. **R12** is not applicable due to the hypothesis on $fc(\{a\}) \sqsubseteq \exists r.\{b\}$;

Together we know that $fc(\{a\}) \sqsubseteq \{b\} \notin ST_{k+1}$.

$fc(\{a\}) \sqsubseteq \exists r.A$ can be derived by **R3, 7, 8**:

1. **R3** is not applicable due to the hypothesis on $fc(\{a\}) \sqsubseteq A$, $fc(\{a\}) \sqsubseteq \{b\}$ and $fc(\{a\}) \sqsubseteq \exists r.A$;
2. **R7** is not applicable due to the hypothesis on $fc(\{a\}) \sqsubseteq \exists r.A$;
3. **R8** is not applicable due to the hypothesis on $fc(\{a\}) \sqsubseteq \exists r.A$, $fc(\{a\}) \sqsubseteq \exists r.\{b\}$ and $fc(\{a\}) \sqsubseteq \exists r.fc(\{b\})$.

Together we know that $fc(\{a\}) \sqsubseteq \exists r.A \notin ST_{k+1}$.

$fc(\{a\}) \sqsubseteq \exists r.\{b\} \notin ST_{k+1}$ and $fc(\{a\}) \sqsubseteq \exists r.fc(\{b\}) \notin ST_{k+1}$ can be proved in a similar manner as above.

Now we have proved that all the claims hold in ST_{k+1} if they hold in ST_k. Combined with the fact that they hold in ST_0, by induction we know that they all hold in $ST_n = ST$. Hence we have $A \sqsubseteq \{a\} \notin ST$. This shows that $\mathcal{O}$ is a semantic nominal-safe ontology under approximate reasoning and proves the theorem. □

Proposition 7 ***(Linear Transformation)*** *In Def. 8, the transformation from $\mathcal{O}$ to $A_{fa}(\mathcal{O})$ can be done in linear time.*

Proof We prove the proposition by showing that for any ontology $\mathcal{O} = (\mathcal{T}_\mathcal{O}, \mathcal{A}_\mathcal{O})$ and its transformation $(\mathcal{T}, \mathcal{A}, CT, QT, IT)$, if $\mathcal{O}$ contains $n_\mathcal{O}$ number of terms, then we have:

1. $|\mathcal{T}| \leq n_\mathcal{O} + |\mathcal{T}_\mathcal{O}|$;
2. $|\mathcal{A}| = |\mathcal{A}_\mathcal{O}|$;
3. $|CT| = n_\mathcal{O}$;
4. $|QT| \leq n_\mathcal{O}$;
5. $|IT| = 4 \times |RN_\mathcal{O}|$.

where $|\mathcal{T}|(|\mathcal{A}|, |\mathcal{T}_\mathcal{O}|, |\mathcal{A}_\mathcal{O}|)$ is the number of axioms in $\mathcal{T}(\mathcal{A}, \mathcal{T}_\mathcal{O}, \mathcal{A}_\mathcal{O})$ and $|CT|$ $(|QT|, |IT|, |\mathcal{RN}_\mathcal{O}|)$ is the number of elements in CT $(QT, IT, \mathcal{RN}_\mathcal{O})$.

The size of $\mathcal{T}$, CT, QT and IT directly follow from proposition 1.

Regarding the size of $\mathcal{A}$, for each axiom in $\mathcal{A}_\mathcal{O}$, at most one axiom is added into $\mathcal{A}$ (Step-2 to 6). □

Theorem 8 ***(Ontology Reasoning Preservation)*** *For any ontology $\mathcal{O}$ and its transformation $(\mathcal{T}, \mathcal{A}, CT, QT, IT)$, let $\mathcal{O}' = (\mathcal{T}', \mathcal{A}')$ be an ontology constructed as follows:*

1. *$\mathcal{T}'$ is constructed in the same way as in Theorem 2;*
2. *$\mathcal{A}' = \mathcal{A}$;*

then for any axiom α with $\Sigma_\alpha \subseteq \Sigma_\mathcal{O}$, we have $\mathcal{O} \models \alpha$ iff $\mathcal{O}' \models \alpha$:

Proof Similar as Theorem 2, this theorem can be proved by showing that $\mathcal{O} \cup \mathcal{O}' \cup \mathcal{T}_{fa}$ conservatively extends both $\mathcal{O}$ and $\mathcal{O}'$ on $\Sigma_\mathcal{O}$, where $\mathcal{T}_{fa}$ is the set of definition $fa(P) \equiv P$ for all term P of $\mathcal{O}$ except complement of roles. Here we slightly extend the notion of conservation extension (CE) to the entire ontology, i.e. an ontology $\mathcal{O}_1$ is a Σ-CE of another ontology $\mathcal{O}_2$ for $\Sigma \subseteq \Sigma_{\mathcal{O}_2}$ iff $\mathcal{O}_2 \subseteq \mathcal{O}_1$ and for any axiom α with $\Sigma_\alpha \subseteq \Sigma$, $\mathcal{O}_1 \models \alpha$ implies $\mathcal{O}_2 \models \alpha$.

The proof is also similar to the proof of Theorem 2. We first show that $\mathcal{O} \cup \mathcal{T}_{fa} \cup \mathcal{O}'$ is a CE of $\mathcal{O}$ by induction. Apparently $\mathcal{O} \subseteq \mathcal{O} \cup \mathcal{O}' \cup \mathcal{T}_{fa}$ and $\mathcal{O} \cup \mathcal{T}_{fa}$ is a CE of $\mathcal{O}$ since it only extends with definitions of fresh names. Assuming some $\mathcal{O}_i$ with $\mathcal{O} \cup \mathcal{T}_{fa} \subseteq \mathcal{O}_i \subseteq \mathcal{O} \cup \mathcal{T}_{fa} \cup \mathcal{T}'$ is a CE of $\mathcal{O}$, and axiom $\beta \in \mathcal{O}' \setminus \mathcal{O}_i$ can either be a TBox axiom or an ABox axiom. When β is a TBox axiom, it can be shown similar as in the proof of Theorem 2 that $\mathcal{O}_i \cup \{\beta\}$ is still a CE of $\mathcal{O}$. When β is an ABox axiom, we consider the following situations:

1. If β is a concept assertion axiom, it is apparent that $\mathcal{O}_i \models \beta$ because β is an ABox axiom in $\mathcal{O}$ with the predicate replaced by its name, which is defined in $\mathcal{T}_{fa}$. Hence $\mathcal{O}_i \cup \{\beta\}$ is still a CE of $\mathcal{O}$;
2. If β is a role assertion axiom $(a, b) : r \in \mathcal{O}'$. There are two possibilities:
   (a) $r = fa(R)$, then there is some $(a, b) : R \in \mathcal{O}$ and $r \equiv R \in \mathcal{T}_{fa}$. Together we have $\mathcal{O}_i \models \beta$ and hence $\mathcal{O}_i \cup \{\beta\}$ is still a CE of $\mathcal{O}$;
   (b) $r = fa(\neg R)$, then there is some $(a, b) : \neg R \in \mathcal{O}$ and possibly $Dis(R, r) \in \mathcal{O}_i$. Since r is a name that do not appear in $\mathcal{O}$, we know that $\mathcal{O}' \cup \beta$ is still a CE of $\mathcal{O}$.

Note that it is impossible for β to be an individual equality or inequality axiom because all such axioms are in $\mathcal{O} \subseteq \mathcal{O}_i$. Together we can show that $\mathcal{O} \cup \mathcal{T}_{fa} \cup \mathcal{O}'$ is a CE of $\mathcal{O}$.

We can also show that $\mathcal{O} \cup \mathcal{T}_{fa} \cup \mathcal{O}'$ is a CE of $\mathcal{O}'$ by showing that $\mathcal{O}' \models \mathcal{T}_{fa}$ and $\mathcal{O}' \cup \mathcal{T}_{fa} \models \mathcal{O}$. The former can be proved in the same way as in the proof of Theorem 2. To prove the latter, let $\beta \in \mathcal{O}$, β can be either a TBox axiom or an ABox axiom. When β is a TBox axiom, we can still show in the same way as in the proof of Theorem 2 that $\mathcal{O}' \cup \mathcal{T}_{fa} \models \beta$. When β is an ABox axiom:

1. If β is an individual equality or inequality axiom, then $\beta \in \mathcal{O}'$ and it's apparent $\mathcal{O}' \cup \mathcal{T}_{fa} \models \beta$.
2. If β is a concept assertion or role assertion axiom, it is apparent that $\mathcal{O}' \cup \mathcal{T}_{fa} \models \beta$ because $\mathcal{O}'$ contains β with expressions replaced by names and $\mathcal{T}_{fa}$ contains the equivalence between expressions and corresponding names;
3. If β is a negative role assertion $(a, b) : \neg R$, by Step-4 of Def. 8 we have $(a, b) : fa(\neg R) \in \mathcal{O}'$. By Def. 4 we also have $(fa(R), fa(\neg R)) \in CT$. Combined with Step-3 of Theorem 2 we have $Dis(fa(R), fa(\neg R)) \in \mathcal{O}'$. We also have $fa(R) \equiv R \in \mathcal{T}_{fa}$. Together we have $\mathcal{O}' \cup \mathcal{T}_{fa} \models \beta$.

From the above we have $\mathcal{O}' \cup \mathcal{T}_{fa} \models \mathcal{O}$. Since we also have $\mathcal{O}' \models \mathcal{T}_{fa}$ we have $\mathcal{O}' \models \mathcal{T}_{fa} \cup \mathcal{O}$. Hence we know that $\mathcal{O}' \cup \mathcal{T}_{fa} \cup \mathcal{O}$ is a CE of $\mathcal{O}'$.

Note that $\Sigma_{\mathcal{O}} \subseteq \Sigma_{\mathcal{O}'}$. Together with the conclusion that $\mathcal{O}' \cup \mathcal{T}_{fa} \cup \mathcal{O}$ is also a CE of $\mathcal{O}$, we can infer hat for any axiom α with $\Sigma_\alpha \subseteq \Sigma_{\mathcal{O}}$, we have $\mathcal{O} \models \alpha$ iff $\mathcal{O}' \cup \mathcal{T}_{fa} \cup \mathcal{O} \models \alpha$ iff $\mathcal{O}' \models \alpha$. □

Proposition 9 ***(Approximation)*** *For an Ontology $\mathcal{O}$, let its transformation results $A_{fa}(\mathcal{O})$ be $(\mathcal{T}, \mathcal{A}, CT, QT, IT)$ as specified in Def. 8, then $(\mathcal{T}, \mathcal{A})$ is an $\mathcal{EL}^{++}$ ontology such that for any axiom α with $\Sigma_\alpha \subseteq \Sigma_{\mathcal{O}}$, we have $(\mathcal{T}, \mathcal{A}) \models \alpha$ only if $\mathcal{O} \models \alpha$.*

Proof It is obvious that $\mathcal{T}$ only contains $\mathcal{EL}^{++}$ TBox axioms and $\mathcal{A}$ only contains $\mathcal{EL}^{++}$ ABox axioms.

The soundness of $(\mathcal{T}, \mathcal{A})$ w.r.t. $\mathcal{O}$ is an apparent corollary of Theorem 8 since $(\mathcal{T}, \mathcal{A}) \subseteq \mathcal{O}'$, which preserves all semantics of $\mathcal{O}$. □

Theorem 10 ***(Complexity)*** *Given an ontology $\mathcal{O}$, its approximate reasoning closure via internalisation $SI_{fa}(\mathcal{O})$ can be computed in polynomial time w.r.t. the size of $\mathcal{O}$.*

Proof According to Def. 10, we have $SI_{fa}(\mathcal{O}) = ST(AI(A_{fa}(\mathcal{O})))$. Assuming $AI(A_{fa}(\mathcal{O})) = (\mathcal{T}, CT, QT, IT)$, from the proof of Theorem 4 we know that $SI_{fa}(\mathcal{O})$ can be computed in polynomial time w.r.t. $|\mathcal{CN}_{\mathcal{T}}| + |\mathcal{RN}_{\mathcal{T}}|$. Thus it is sufficient to prove that:

1. $AI(A_{fa}(\mathcal{O}))$ can be computed in polynomial time w.r.t. the size of $\mathcal{O}$;
2. $|\mathcal{CN}_{\mathcal{T}}| + |\mathcal{RN}_{\mathcal{T}}|$ is polynomial w.r.t. the size of $\mathcal{O}$.

These two propositions are quite obvious because $A_{fa}(\mathcal{O})$ can be computed in linear time w.r.t. the size of $\mathcal{O}$ and its size is also linear w.r.t. the size of $\mathcal{O}$ (Proposition 7), and from Def. 9 we know that $AI(A_{fa}(\mathcal{O}))$ can be computed in linear time w.r.t. the size of $A_{fa}(\mathcal{O})$ and the size of the former is also linear w.r.t. the size of the later. The theorem can thus be proved. □

Theorem 11 ***(Approximate Reasoning via Internalisation Soundness)*** *For an ontology $\mathcal{O}$, let α be an axiom with $\Sigma_\alpha \subseteq \Sigma_{\mathcal{O}}$, then $\alpha \in SI_{fa}(\mathcal{O})$ only if $\mathcal{O} \models \alpha$.*

Proof Let $\mathcal{O} = (\mathcal{T}, \mathcal{A})$, where $\mathcal{T}$ and $\mathcal{A}$ are TBox and ABox of $\mathcal{O}$, respectively. We construct a new ontology $\mathcal{O}' = (\mathcal{T}', \emptyset)$ as follows:

1. $\mathcal{T}$ is initialised as $\mathcal{T}$;
2. For each $a \in \mathcal{IN}_{\mathcal{O}}, \mathcal{T}' = \mathcal{T}' \cup \{\{a\} \sqsubseteq \{a\}, \neg\{a\} \sqsubseteq \neg\{a\}, \{a\} \sqsubseteq \top, \neg\{a\} \sqsubseteq \top\}$
3. For each $a : C \in \mathcal{A}$, $\mathcal{T}' = \mathcal{T}' \cup \{\{a\} \sqsubseteq C\}$;
4. For each $(a, b) : R \in \mathcal{A}$, $\mathcal{T}' = \mathcal{T}' \cup \{\{a\} \sqsubseteq \exists R.\{b\}\}$;
5. For each $(a, b) : \neg R \in \mathcal{A}$, $\mathcal{T}' = \mathcal{T}' \cup \{\{a\} \sqsubseteq \exists fa(\neg R).\{b\}, Dis(R, fa(\neg R))\}$;
6. For each $a \doteq b \in \mathcal{A}$, $\mathcal{T}' = \mathcal{T}' \cup \{\{a\} \equiv \{b\}\}$;
7. For each $a \not\doteq b \in \mathcal{A}$, $\mathcal{T}' = \mathcal{T}' \cup \{\{a\} \sqsubseteq\sim\{b\}\}$;

Apparently $\mathcal{O}'$ is semantically equivalent to $\mathcal{O}$ w.r.t. entailment of any axiom α constructed with $\Sigma_{\mathcal{O}}$. In other words, $\mathcal{O}' \models \alpha$ *iff* $\mathcal{O} \models \alpha$.

From Def. 4, Def. 8 and Def. 9 it is obvious that $AI(A_{fa}(\mathcal{O})) = TA_{fa}(\mathcal{O}')$. Thus $SI_{fa}(\mathcal{O}) = ST(AI(A_{fa}(\mathcal{O}))) = ST(TA_{fa}(\mathcal{O}'))$. According to Theorem 5 we know that $ST(TA_{fa}(\mathcal{O}'))$ is soundness-preserving, thus $SI_{fa}(\mathcal{O})$ should also be soundness-preserving. This proves the theorem. □

Theorem 12 ***(Concept Subsumption Checking)*** *For a nominal-free ontology $\mathcal{O}$, let $a, b \in \mathcal{IN}_{\mathcal{O}}$ be two individuals, $A, B \in \mathcal{CN}_{\mathcal{O}}$ be two concepts, and $r \in \mathcal{RN}_{\mathcal{O}}$ be a role, then the following holds:*

1. $A \sqsubseteq B \in SI_{fa}(\mathcal{O})$ iff $A \sqsubseteq B \in SNF_{fa}(\mathcal{O})$;
2. $\{a\} \sqsubseteq A \in SI_{fa}(\mathcal{O})$ iff $a : A \in SNF_{fa}(\mathcal{O})$;
3. $\{a\} \sqsubseteq \exists r.\{b\} \in SI_{fa}(\mathcal{O})$ iff $(a, b) : r \in SNF_{fa}(\mathcal{O})$;

Proof The ← direction is obvious because the **AR** rules are special cases of the corresponding **R** rules where certain atomic concepts are restricted to nominals. If we rewrite the ABox axioms into corresponding nominal form TBox axioms, the **AR** rules are the same as or can be reduced to combinations of **R** rules. Some rules require further explanations:

1. The premises of **AR1c** rule are reduced to $\{x\} \sqsubseteq A$ and $\{y\} \sqsubseteq fc(A)$, which can derive their consequence $\{x\} \sqsubseteq fc(\{y\})$ by combination of **R1** and **R10**.
2. The **AR1e** rule seems unnecessary but it is to derive $\{a\} \sqsubseteq \{b\}$ from $\{a\} \sqsubseteq \bot$ and $\bot \sqsubseteq \{b\}$, which does not have an ABox counterpart form.
3. When $\sharp = \doteq$, the **AR10** rule is an apparent tautology, due to internalisation of individual equality in Step-5 of Def. 9;

4. When $\sharp = \neq$, the **AR10** rule is reduced to **R10** due to the internalisation of individual inequality in Step-6 of Def. 9;

Let $A_{fa}(\mathcal{O}) = (\mathcal{T}, \mathcal{A}, CT, QT, IT)$ and $AI(A_{fa}(\mathcal{O})) = \{\mathcal{T}', CT, QT, IT\}$. In the following, we use SI to denote $SI_{fa}(\mathcal{O})$, ST to denote $ST((\mathcal{T}, CT, QT, IT))$, and SNF to denote $SNF_{fa}(\mathcal{O})$. Note that when the ontology $\mathcal{O}$ is nominal free, $\mathcal{T}$ does not contain any nominal or names for complement of nominal. Hence $\mathcal{CN}_\mathcal{T}$ also does not include names for complement of nominal.

Before we prove the → direction, we analyse the forms of axioms in SI. Because the ontology is nominal-free, all axioms in the initial $\mathcal{T}'$ should not have $fc(\{a\})$ as the LHS unless the RHS is $\top$ or also some $fc(\{b\})$, and should not have $\{a\}$ as the RHS unless the LHS is $\bot$ or also some nominal.

Also, most of the **R** rules that are applicable on nominal-free ontology have the following features: if the rule derives a consequence of form $A \sqsubseteq B$, where A and B are either atomic concepts or nominals, then there is a premise axiom with B as the RHS, and a premise axiom with A as the LHS. The only exceptions are the **R10** which derive consequence with LHS and RHS not appearing in premises, and the **R11**, where the consequence has RHS not appearing in premise. However, to derive $\{a\}$ as the RHS of consequence by **R10**, it is required to have a premise with $fc(\{a\})$ as the LHS, and vice versa. The same applies to deriving $\{a\}$ as the RHS of consequence by **R11**. Thus by induction, we know that for a nominal-free ontology $\mathcal{O}$, SI have the following properties, where A, B are named concepts or $\top$, $\bot$, and a, b are individuals:

1. $A \sqsubseteq \{a\} \notin SI$ unless $A \sqsubseteq \bot \in SI$;
2. $fc(\{a\}) \sqsubseteq B \notin SI$ unless $\top \sqsubseteq B \in SI$;
3. $fc(\{a\}) \sqsubseteq \{b\} \notin SI$;

Similarly, we know that SI have the following properties when the ontology is nominal-free:

1. $A \sqsubseteq \exists r.\{a\} \notin SI$ unless $A \sqsubseteq \bot \in SI$;
2. $A \sqsubseteq \exists r.fc(\{a\}) \notin SI$ unless $A \sqsubseteq \bot \in SI$;
3. $\{a\} \sqsubseteq \exists r.fc(\{a\}) \notin SI$;
4. $fc(\{a\}) \sqsubseteq \exists r.A \notin SI$;
5. $fc(\{a\}) \sqsubseteq \exists r.\{b\} \notin SI$;
6. $fc(\{a\}) \sqsubseteq \exists r.fc(\{b\}) \notin SI$;

Furthermore, because ontology is nominal-free, rule **R18** is no longer applicable. Thus all RIs in SI are independent to GCIs and ABox assertions. So for any RI β, we have $\beta \in SI$ *iff* $\beta \in ST$.

Similarly, because we have $A \sqsubseteq \top \in \mathcal{T}$ for all $A \in \mathcal{CN}_\mathcal{T}$, we also have $A \sqsubseteq \top \in ST$ and $\bot \sqsubseteq A \in ST$.

Thus we only need to consider the other kind of entailments that contains nominals in SI.

We prove the → directions of the theorem by proving the following stronger lemma:

Lemma 13 *For a nominal-free ontology $\mathcal{O}$, let $A_{fa}(\mathcal{O}) = (\mathcal{T}, \mathcal{A}, CT, QT, IT)$ and $AI(A_{fa}(\mathcal{O}))$ be $(\mathcal{T}', CT, QT, IT)$. If we use SI to denote $SI_{fa}(\mathcal{O})$, ST to denote $ST((\mathcal{T}, CT, QT, IT))$, and SNF to denote $SNF_{fa}(\mathcal{O})$, then the following invariants hold in each step of computation of SI, where $a, b \in \mathcal{IN}_\mathcal{O}$ are two individuals, $A, B \in \mathcal{CN}_\mathcal{T} \cup \{\top, \bot\}$ are two concepts, and*

$r \in \mathcal{RN}_\mathcal{O}$ *is a role:*

$$A \sqsubseteq B \in SI \textit{ only if } A \sqsubseteq B \in ST \tag{36}$$

$$A \sqsubseteq fc(\{a\}) \in SI \textit{ only if } a : fc(A) \in SNF \tag{37}$$

$$\{a\} \sqsubseteq B \in SI \textit{ only if } a : B \in SNF \tag{38}$$

$$\{a\} \sqsubseteq \{b\} \in SI \textit{ only if } a \doteq b \in SNF \tag{39}$$

$$\{a\} \sqsubseteq fc(\{b\}) \in SI \textit{ only if } a \not\doteq b \in SNF \tag{40}$$

$$fc(\{a\}) \sqsubseteq fc(\{b\}) \in SI \textit{ only if } a \doteq b \in SNF \tag{41}$$

$$A \sqsubseteq \bot \notin SI \textit{ and } A \sqsubseteq \exists r.B \in SI \textit{ only if } A \sqsubseteq \exists r.B \in ST \tag{42}$$

$$\{a\} \sqsubseteq \bot \notin SI \textit{ and } \{a\} \sqsubseteq \exists r.B \in SI \textit{ only if } a : \exists r.B \in SNF \tag{43}$$

$$\{a\} \sqsubseteq \exists r.\{b\} \in SI \textit{ only if } (a,b) : r \in SNF \tag{44}$$

Apparently the → directions of the theorem hold if the above lemma holds.

To prove the above invariants, we perform induction of the application of rules. First of all, it is obvious that the invariants are satisfied by all initial elements of SI, i.e. all axioms in $\mathcal{T}'$ and corresponding tautologies added into $SI = ST((\mathcal{T}', CT, QT, IT))$ in Step-1 of Def. 5.

Then we show that, the above invariants hold for each of the corresponding axioms derived by each of the corresponding rules.

Invariant 36: $A \sqsubseteq B \in SI$ can be derived by **R1, 2, 4, 5, 9, 10, 11, 13**. Note that **R12** can not derive $A \sqsubseteq B \in SI$ when the original ontology $\mathcal{O}$ is nominal-free. We make a case distinction as follows:

1. If it is derived by **R1**. Then there is a concept X s.t. $A \sqsubseteq X$ and $X \sqsubseteq B$ in SI. We make a case distinction on the form of X:
   (a) If $X \in \mathcal{CN}_\mathcal{T}$, by induction of invariant 36 we know that $A \sqsubseteq X, X \sqsubseteq B \in ST$. Thus by non-applicability of rule **R1** we have $A \sqsubseteq B \in ST$.
   (b) If X is a nominal $\{a\}$, then we have $A \sqsubseteq \{a\} \in SI$, which is contradictory to the form of axioms appearing in SI unless $A \sqsubseteq \bot \in SI$. By induction of invariant 36 we have $A \sqsubseteq \bot \in ST$. Together with $\bot \sqsubseteq B \in ST$ we directly have $A \sqsubseteq B \in ST$.
   (c) If X is of form $fc(\{b\})$, then we have $fc(\{b\}) \sqsubseteq B \in SI$, which is contradictory to the form of axioms in appearing in SI unless $\top \sqsubseteq B \in SI$. By induction of invariant 36 we have $\top \sqsubseteq B \in ST$. Together with $A \sqsubseteq \top \in ST$ we directly have $A \sqsubseteq B \in ST$.
2. If it is derived by **R2**, then there are concepts $A_1, \ldots, A_n$ s.t. $A \sqsubseteq A_i \in SI$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq B \in \mathcal{T}'$. Given the form of the later, it should also be in $\mathcal{T}$. And also $A_i \in \mathcal{CN}_\mathcal{T}$. In this case, by induction of invariant 36 and the non-applicability of **R2** we have $A \sqsubseteq B \in ST$.
3. If it is derived by **R4**, then there is $r \in \mathcal{RN}'_\mathcal{T}$ and two concepts X and X' such that $A \sqsubseteq \exists r.X \in SI$, $X \sqsubseteq X' \in SI$ and $\exists r.X' \sqsubseteq B \in \mathcal{T}'$. Given the form of $\exists r.X' \sqsubseteq B$, it should also be in $\mathcal{T}$ and X' must be in $\mathcal{CN}_\mathcal{T}$. We make a case distinction on the form of X:
   (a) If $X \in \mathcal{CN}_\mathcal{T}$, when $A \sqsubseteq \bot \in SI$, then by induction of invariant 36 we have $A \sqsubseteq \bot \in ST$ and hence $A \sqsubseteq B \in ST$. Otherwise, by induction of invariant 42 we know that $A \sqsubseteq \exists r.X \in ST$. By induction of invariant 36 we also know that $X \sqsubseteq X' \in ST$. Thus by non-applicability of rule **R4** we have $A \sqsubseteq B \in ST$.

(b) If x is a nominal $\{b\}$, the we have $A \sqsubseteq \exists r.\{b\} \in SI$, which contradicts the possible forms of axioms appearing in SI except $A \sqsubseteq \bot \in SI$. By induction of invariant 36 we have $A \sqsubseteq \bot \in ST$. Hence we will have $A \sqsubseteq B \in ST$.

(c) If X is of form $fc(\{b\})$, we skip this situation as it is similar as the above.

4. If it is derived by **R5**, then $B = \bot$ and there is concept X and $r \in \mathcal{RN}_\mathcal{T}$ s.t. $A \sqsubseteq \exists r.X \in SI$ and $X \sqsubseteq \bot \in SI$. We make a case distinction on the form of X:
   (a) If $X \in \mathcal{CN}_\mathcal{T}$. When $A \sqsubseteq \bot \in SI$ we trivially have $A \sqsubseteq B \in ST$ as before. Otherwise by invariant 42 we have $A \sqsubseteq \exists r.X \in ST$ and $X \sqsubseteq \bot \in ST$. Due to the non-applicability of **R5** we have $A \sqsubseteq B \in ST$.
   (b) If X is a nominal $\{a\}$, then we have $A \sqsubseteq \exists r.\{a\} \in SI$. Given the possible forms of axioms in SI, we have $A \sqsubseteq \bot \in SI$. Hence by induction of invariant 36 we have $A \sqsubseteq B \in ST$.
   (c) If X is of form $fc(\{a\})$, then we have $fc(\{a\}) \sqsubseteq \bot \in SI$, which is contradictory to the possible forms of axioms in SI thus we omit this situation.

   Considering the form of $A \sqsubseteq \exists r.X$, we know that
5. If it is derived by **R9**, then $B = \bot$ and there is a concept X s.t. $A \sqsubseteq X$ and $A \sqsubseteq fc(X)$. We make a case distinction on form of X:
   (a) If $X \in \mathcal{CN}_\mathcal{T}$, thus by invariant 36 we have $A \sqsubseteq X, A \sqsubseteq fc(X) \in ST$. Due to the non-applicability of **R9** we have $A \sqsubseteq \bot \in ST$.
   (b) If X is a nominal $\{a\}$, then we have $A \sqsubseteq \{a\} \in SI$. Given the possible form of axioms in SI, we have $A \sqsubseteq \bot \in SI$. By induction of invariant 36 we directly have $A \sqsubseteq B \in ST$.
   (c) If X is of form $fc(\{a\})$, then we have $A \sqsubseteq fc(fc(\{a\})) \in SI$, i.e. $A \sqsubseteq \{a\} \in SI$. Then we skip this situation as it is the same as above.
6. If it is derived by **R10**, then we have $fc(B) \sqsubseteq fc(A) \in SI$. Because $A, B \in \mathcal{CN}_\mathcal{T}$, $fc(A)$ and $fc(B)$ are also in $\mathcal{CN}_\mathcal{T}$. By induction on invariant 36 we have $fc(B) \sqsubseteq fc(A) \in ST$. Due to the non-applicability of rule **R10** we have $A \sqsubseteq B \in ST$.
7. If it is derived by **R11**, then we have $A_1, \ldots, A_n$ s.t. $B = fc(A_i)$ and $A \sqsubseteq A_j \in SI(1 \leq j \leq n, j \neq i)$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq \bot \in \mathcal{T}$. Given the possible forms of axioms in SI, we know that $A_1, \ldots, A_n \in \mathcal{CN}_\mathcal{T}$. Due to invariant 36 we have $A \sqsubseteq A_j \in ST$ $(1 \leq j \leq n, j \neq i)$. Thus by non-applicability of rule **R11** we have $A \sqsubseteq B \in ST$.
8. If it is derived by **R13**, then by induction of invariant 36 it is obvious that $A \sqsubseteq B \in ST$ because the internalisation does not change QT.

Thus the induction of invariant 36 is proved.

Invariant 37: $A \sqsubseteq fc(\{a\}) \in SI$ can be derived by **R1, 10**. Note that **R2** can not derive $A \sqsubseteq fc(\{a\})$ because in $\mathcal{T}'$ there is no GCI with $fc(\{a\})$ as the RHS and a conjunction as the LHS. Similarly, **R4** can not derive $A \sqsubseteq fc(\{a\})$.

We skip the obvious situation where $A = \bot$ and make a case distinction as follows:

1. If it is derived by **R1**, then there is a concept X s.t. $A \sqsubseteq X \in SI$ and $X \sqsubseteq fc(\{a\}) \in SI$. We make a case distinction on the form of X:
   (a) If $X \in \mathcal{CN}_\mathcal{T}$, then by induction of invariant 36 we have $A \sqsubseteq X \in ST$. Due to the non-applicability of **R10** we have $fc(X) \sqsubseteq fc(A) \in ST \subseteq SNF$. By induction of invariant 37 we have $a : fc(X) \in SNF$. Due to the non-applicability of **AR1a** we have $a : fc(A) \in SNF$.

(b) If X is a nominal $\{b\}$, then we have $A \sqsubseteq \{b\} \in SI$, which is contradictory to the possible forms of axioms in SI unless we have $A \sqsubseteq \bot \in SI$. By induction of invariant 36 we have $A \sqsubseteq \bot \in ST$. Due to the non-applicability of **R10** we also have $\top \sqsubseteq fc(A) \in ST \subseteq SNF$. Given that $a : \top \in SNF$, due to the non-applicability of **AR1a** we have $a : fc(A) \in SNF$.

(c) If X is of form $fc(\{b\})$, then by induction of invariant 37, we have $b : fc(A) \in SNF$. By induction of invariant 41 we have $a \doteq b \in SNF$. Due to the non-applicability of **AR1b** we have $a : fc(A) \in SNF$.

2. If it is derived by **R10**, then there is $\{a\} \sqsubseteq fc(A) \in SI$. By induction of invariant 38 we have $a : fc(A) \in SNF$.

Thus the induction of invariant 37 is proved.

Invariant 38: $\{a\} \sqsubseteq B \in SI$ can be derived by **R1, 2, 4, 5, 9, 10, 11**.

We skip the obvious situation where $B = \top$ and make a case distinction as follows:

1. If it is derived by **R1**. Then there is a concept X s.t. $\{a\} \sqsubseteq X$ and $X \sqsubseteq B$. We make a case distinction on the form of X:
   (a) If $X \in \mathcal{CN}_{\mathcal{T}}$, by induction of invariant 38 we know that $a : X \in SNF$. By induction of invariant 36 we also know that $X \sqsubseteq B \in ST \subseteq SNF$. Thus by non-applicability of rule **AR1a** we have $a : B \in SNF$.
   (b) If X is a nominal $\{b\}$, by induction of invariant 39 we have $a \doteq b \in SNF$. By induction of invariant 38 we have $b : B \in SNF$. Together by non-applicability of rule **AR1b** we have $a : B \in SNF$.
   (c) If X is of form $fc(\{b\})$, then we have $fc(\{b\}) \sqsubseteq B \in SI$, which is contradictory to the form of axioms appearing in SI unless we have $\top \sqsubseteq B \in SI$. By induction of invariant 36 we have $\top \sqsubseteq B \in ST \subseteq SNF$. Given that $a : \top \in SNF$, due to the non-applicability of **AR1a** we have $a : B \in SNF$.
2. If it is derived by **R2**, then there are concepts $A_1, \ldots, A_n$ s.t. $\{a\} \sqsubseteq A_i \in SI$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq B \in \mathcal{T}'$. Given the form of the later, it should also be in $\mathcal{T}$. And also $A_i \in \mathcal{CN}_{\mathcal{T}}$. In this case, by induction of invariant 38 and the non-applicability of **AR2** we have $x : B \in SNF$.
3. If it is derived by **R4**, then there is $r \in \mathcal{RN}_{\mathcal{T}}$ and two concepts X and X' such that $\{a\} \sqsubseteq \exists r.X \in SI$, $X \sqsubseteq X' \in SI$ and $\exists r.X' \sqsubseteq B \in \mathcal{T}'$. Given the form of $\exists r.X' \sqsubseteq B$, it should also be in $\mathcal{T}$ and X' must be in $\mathcal{CN}_{\mathcal{T}}$. We skip the obvious situation where $\{a\} \sqsubseteq \bot \in SI$ before **R4** is applied. We make a case distinction on the form of X:
   (a) If $X \in \mathcal{CN}_{\mathcal{T}}$, by induction of invariant 43 we know that $a : \exists r.X \in SNF$. By induction of invariant 36 we also know that $X \sqsubseteq X' \in ST \subseteq SNF$. Thus by non-applicability of rule **AR4a** we have $a : B \in SNF$.
   (b) If x is a nominal $\{b\}$, by induction of invariant 44 we have $(a, b) : r \in SNF$. By induction of invariant 38 we have $b : X' \in SNF$. Thus by non-applicability of rule **AR4b** we have $a : B \in SNF$.
   (c) If X is of form $fc(\{b\})$, then we have $\{a\} \sqsubseteq \exists r.fc(\{b\}) \in SI$, which is contradictory to the form of axioms appearing in SI thus we omit this situation.
4. If it is derived by **R5**, then $B = \bot$ and there are concept X and $r \in \mathcal{RN}_{\mathcal{T}}$ s.t. $\{a\} \sqsubseteq \exists r.X \in SI$ and $X \sqsubseteq \bot \in SI$. Similar as above, we skip the obvious situation where $\{a\} \sqsubseteq \bot \in SI$ before **R5** is applied. We make a case distinction on the form of X:

   (a) If $X \in \mathcal{CN}_\mathcal{T}$, then by induction of invariant 43 we have $\{a\} \sqsubseteq \exists r.X \in SNF$. By induction of invariant 36 we have $X \sqsubseteq \bot \in ST \subseteq SNF$. Due to the non-applicability of rule **AR5a** we have $a : \bot \in SNF$.
   (b) If X is a nominal $\{b\}$, they by induction of invariant 44 we have $(a, b) : r \in SNF$. By induction of invariant 38 we also have $b : \bot \in SNF$. Due to the non-applicability of rule **AR5b** we have $a : \bot \in SNF$.
   (c) If X is of form $fc(\{b\})$, it is contradictory to the possible forms of axioms in SI thus we omit this situation.
5. If it is derived by **R9**, then $B = \bot$ and there is a concept X s.t. $\{a\} \sqsubseteq X \in SI$ and $\{a\} \sqsubseteq fc(X) \in SI$. We make a case distinction on the form of X:
   (a) If $X \in \mathcal{CN}_\mathcal{T}$, then by induction of invariant 38 we have $a : X \in SNF$. Similarly we have $a : fc(X) \in SNF$. Due to the non-applicability of **AR9a** we have $a : \bot \in SNF$.
   (b) If X is a nominal $\{b\}$, then by induction of invariant 39 we have $a \doteq b \in SNF$. By induction of invariant 40 we also have $a \not\doteq b \in SNF$. Due to the non-applicability of **AR9b** we have $a : \bot \in SNF$.
   (c) If X is of form $fc(\{b\})$, then similar as the above situation, we have $a : \bot \in SNF$.
6. If it is derived by **R10**, then we have $fc(B) \sqsubseteq fc(\{a\}) \in SI$. By induction of invariant 37 we have $a : B \in SNF$.
7. If it is derived by **R11**, then we have $A_1, \ldots, A_n$ s.t. $B = fc(A_i)$ and $\{a\} \sqsubseteq A_j \in SI$ $(2 \leq j \leq n, j \neq i)$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq \bot \in \mathcal{T}$. Given the possible forms of axioms in SI, we know that $A_1, \ldots, A_n \in \mathcal{CN}_\mathcal{T}$. By induction of invariant 38 we have $a : A_j \in SNF$ $(1 \leq j \leq n, j \neq i)$. Due to the non-applicability of **AR11** we have $a : B \in SNF$.

Thus the induction of invariant 38 is proved.

Invariant 39: $\{a\} \sqsubseteq \{b\}$ can be derived by **R1, 6, 10**. For the similar reason as of $A \sqsubseteq fc(\{a\})$, **R2** and **R4** can not derive $\{a\} \sqsubseteq \{b\}$.

1. If it is derived by **R1**, then there is a concept X s.t. $\{a\} \sqsubseteq X \in SI$ and $X \sqsubseteq \{b\} \in SI$. We make a case distinction on the form of X:
   (a) If $X \in \mathcal{CN}_\mathcal{T}$, given the possible forms of axioms in SI, we have $X \sqsubseteq \bot \in SI$. By induction of invariant 38 we have $a : X \in SNF$. By induction of invariant 36 we have $X \sqsubseteq \bot \in ST \subseteq SNF$. Due to the non-applicability of **AR1a** we have $a : \bot \in SNF$. Due to the non-applicability of **AR1e** we have $a \doteq b \in SNF$.
   (b) If X is a nominal $\{c\}$. Thus by induction of invariant 39 we have $a \doteq c \in SNF$ and $c \doteq b \in SNF$. By non-applicability of **AR1d** we have $a \doteq b \in SNF$.
   (c) If X is of form $fc(\{c\})$, then we have $fc(\{c\}) \sqsubseteq \{b\} \in SI$, which is contradictory to the possible forms of axioms in SI thus we omit this situation.
2. If it is derived by **R6**, then there is $\{b\} \sqsubseteq \{a\} \in SI$. As we explained earlier in the proof, if $\{b\} \sqsubseteq \{a\}$ is an initial element of SI without any inference, then we have $b \doteq a \in SNF$. This is because the original ontology is nominal free so that $\{b\} \sqsubseteq \{a\} \in SI$ can only be included due to either Step-4 or Step-7 of Def. 9, which both leads to $b \doteq a \in SNF$. Alternatively, if $\{b\} \sqsubseteq \{a\} \in SI$ is derived, then by induction of invariant 39 we have $b \doteq a \in SNF$. In either case, by non-applicability of **AR10** we have $a \doteq b \in SNF$.
3. If it is derived by **R10**, then there is $fc(\{b\}) \sqsubseteq fc(\{a\}) \in SI$. By induction of invariant 41 we have $b \doteq a \in SNF$. Due to the non-applicability of **AR10** we have $a \doteq b \in SNF$.

Thus the induction of invariant 39 is proved.

Invariant 40: Similar as above, $\{a\} \sqsubseteq fc(\{b\})$ can be derived by **R1, 10, 12, 20**. In the following, we skip the obvious situation where $\{a\} \sqsubseteq \bot \in SI$ is derived before the corresponding rule is applied, in which case we have $a : \bot \in SNF$ by induction of invariant 38. Then due to the non-applicability of **AR1e** we can derive $a \neq b \in SNF$. We make a case distinction as follows:

1. If it is derived by **R1**, then there is a concept X s.t. $\{a\} \sqsubseteq X \in SI$ and $X \sqsubseteq fc(\{b\}) \in SI$. We make a case distinction on the form of X:
   (a) If $X \in \mathcal{CN}_\mathcal{T}$, then by induction of invariant 38 we have $a : X \in SNF$. By induction of invariant 37 we have $b : fc(X) \in SNF$. Due to non-applicability of **AR1c** we have $a \neq b \in SNF$.
   (b) If X is a nominal $\{c\}$, then by induction of invariant 39 we have $a \doteq c \in SNF$. By induction of invariant 40 we have $c \neq b \in SNF$. By non-applicability of **AR1d** we have $a \neq b \in SNF$.
   (c) If X is of form $fc(\{c\})$, by induction of invariant 40 we have $a \neq c \in SNF$. By induction of invariant 41 we have $c \doteq b \in SNF$. Due to non-applicability of **AR10** and **AR1d** we eventually have $a \neq b \in SNF$.
2. If it is derived by **R10**, then there is $\{b\} \sqsubseteq fc(\{a\}) \in SI$. By induction of invariant 40 we have $b \neq a \in SNF$. Due to the non-applicability of **AR10** we have $a \neq b \in SNF$.
3. If it is derived by **R12**, then there is $r \in \mathcal{RN}_\mathcal{T}$ and $c \in \mathcal{IN}_\mathcal{T}$ s.t. $\{a\} \sqsubseteq \exists r.\{c\}, \{b\} \sqsubseteq \exists fc(r).\{c\} \in SI$. By induction of invariant 44 we have $(a, c) : r \in SNF$ and $(b, c) : fc(r) \in SNF$. Due to the non-applicability of **AR12** we have $x \neq y$.
4. If it is derived by **R20**, then there is concept X s.t. $\{a\} \sqsubseteq \exists r.X, X \sqsubseteq \exists fi(fc(r)).\{b\} \in SI$. We make a case distinction on the form of X:
   (a) If $X \in \mathcal{CN}_\mathcal{T}$, given the possible forms of axioms in SI, we have $X \sqsubseteq \bot \in SI$. By induction of invariant 43 we have $a : \exists r.X \in SNF$. By induction of invariant 36 we have $X \sqsubseteq \bot \in ST \subseteq SNF$. Together due to the non-applicability of **AR5a** we have $a : \bot \in SNF$. Given that we have $b : \top \in SNF$, due to the non-applicability of **AR1c** we have $a \neq b \in SNF$.
   (b) If X is a nominal $\{c\}$, by induction of invariant 44 we have $(a, c) : r \in SNF$ and $(c, b) : fi(fc(r)) \in SNF$. Due to the non-applicability of **AR19** we have $(c, a) : fi(r) \in SNF$. Because $fi(fc(r)) = fc(fi(r))$, due to the non-applicability of **AR12** we have $a \neq b \in SNF$.
   (c) If X is of form $fc(\{c\})$, then we have $\{a\} \sqsubseteq \exists r.fc(\{c\}) \in SI$, which contradicts with the possible forms of axioms in SI thus we omit this situation.

Thus the induction of invariant 40 is proved.

Invariant 41: Similar as above again, $fc(\{a\}) \sqsubseteq fc(\{b\})$ can be derived by **R1, 10**.

1. If it is derived by **R1**, then there is a concept X s.t. $fc(\{a\}) \sqsubseteq X \in SI$ and $X \sqsubseteq fc(\{b\}) \in SI$. We make a case distinction on form of X:
   (a) If $X \in \mathcal{CN}_\mathcal{T}$, given the possible form of axioms in SI, we have $\top \sqsubseteq X \in SI$. By induction of invariant 36 we have $\top \sqsubseteq X \in ST$. Due to the non-applicability of **R10** we have $fc(X) \sqsubseteq \bot \in ST \subseteq SNF$. By induction of invariant 37 we have $b : fc(X) \in SNF$. Together due to the non-applicability of **AR1a** we have $b : \bot \in SNF$. Due to the non-applicability of **AR1e** and **AR10** we eventually have $a \doteq b \in SNF$.

   (b) If X is a nominal $\{c\}$, then we have $fc(\{a\}) \sqsubseteq \{c\} \in SI$, which contradicts with the possible forms of axioms in SI thus we omit this situation.
   (c) If X is of form $fc(\{c\})$. Thus by induction of invariant 41 we have $a \doteq c \in SNF$ and $c \doteq b \in SNF$. By non-applicability of **AR1d** we have $a \doteq c \in SNF$.
2. If it is derived by **R10**, then there is $\{b\} \sqsubseteq \{a\} \in SI$. By induction of invariant 39 we have $b \doteq a \in SNF$. Due to the non-applicability of **AR10** we have $a \doteq b \in SNF$.

Thus the induction of invariant 41 is proved.

Invariant 42: $A \sqsubseteq \exists r.B \in SI$ can be derived by **R3, 7, 8**. Note that $A \sqsubseteq \bot \notin SI$. We make a case distinction as follows:

1. If it is derived by **R3**, then there is a concept X s.t. $A \sqsubseteq X, X \sqsubseteq \exists r.B \in SI$. Given the possible forms of axioms in SI, we only have $X \in \mathcal{CN}_\mathcal{T}$. Then by induction of invariant 36 we have $A \sqsubseteq X \in ST$. By induction of invariant 42 we have $X \sqsubseteq \exists r.B \in ST$. Due to the non-applicability of rule **R3** we have $A \sqsubseteq \exists r.B \in ST$.
2. If it is derived by **R7**, there is $s \in \mathcal{RN}_\mathcal{T}$ s.t. $A \sqsubseteq \exists s.B \in SI$ and $s \sqsubseteq r \in SI$ which should also be in ST. By induction of invariant 42 we have $A \sqsubseteq \exists s.B \in ST$. Due to non-applicability of **R7** we have $A \sqsubseteq \exists r.B \in ST$.
3. If it is derived by **R8**, there are concept X and roles $r_1, r_2 \in \mathcal{RN}_\mathcal{T}$ s.t. $A \sqsubseteq \exists r_1.X, X \sqsubseteq \exists r_2.B, r_1 \circ r_2 \sqsubseteq R \in SI$. Given the possible forms of axioms in SI, we have $X \in \mathcal{CN}_\mathcal{T}$. By induction of invariant 42 we have $A \sqsubseteq \exists r_1.X, X \sqsubseteq \exists r_2.B \in ST$. Due to the non-applicability of **R8** we have $A \sqsubseteq \exists r.B \in ST$.

Thus the induction of invariant 42 is proved.

Invariant 43: Similarly as above, $\{a\} \sqsubseteq \exists r.B \in SI$ can be derived by **R3, 7, 8**. Note that $\{a\} \sqsubseteq \bot \notin SI$. We make a case distinction as follows:

1. If it is derived by **R3**, then there is a concept X s.t. $\{a\} \sqsubseteq X, X \sqsubseteq \exists r.B \in SI$. We make a case distinction on form of X:
   (a) If $X \in \mathcal{CN}_\mathcal{T}$, then by induction of invariant 38 we have $a : X \in SNF$. By induction of invariant 42 we have $X \sqsubseteq \exists r.B \in ST \subseteq SNF$. Due to the non-applicability of rule **AR3a** we have $a : \exists r.B \in SNF$.
   (b) If X is a nominal $\{b\}$, then by induction of invariant 44 we have $a \doteq b \in SNF$. By induction of invariant 43 we have $b : \exists r.B \in SI$. Due to the non-applicability of **AR3b** we have $a : \exists r.B \in SI$.
   (c) If X is of form $fc(\{b\})$, then we have $fc(\{b\}) \sqsubseteq \exists r.B \in SI$, which is contradictory to possible forms of axioms in SI thus we omit this situation.
2. If it is derived by **R7**, there is $s \in \mathcal{RN}_\mathcal{T}$ s.t. $\{a\} \sqsubseteq \exists s.B \in SI$ and $s \sqsubseteq r \in SI$ which should also be in ST. By induction of invariant 43 we have $a : \exists s.B \in SNF$. Due to non-applicability of **AR7a** we have $a : \exists r.B \in SNF$.
3. If it is derived by **R8**, there are concept X and roles $r_1, r_2 \in \mathcal{RN}_\mathcal{T}$ s.t. $\{a\} \sqsubseteq \exists r_1.X, X \sqsubseteq \exists r_2.B, r_1 \circ r_2 \sqsubseteq R \in SI$. We make a case distinction on form of X:
   (a) If $X \in \mathcal{CN}_\mathcal{T}$, by induction of invariant 43 we have $a : \exists r_1.X$. By induction of invariant 42 we have $X \sqsubseteq \exists r_2.B \in ST \subseteq SNF$. Due to the non-applicability of **AR8a** we have $a : \exists r.B \in SNF$.
   (b) If X is a nominal $\{b\}$, by induction of invariant 44 we have $(a, b) : r_1 \in SNF$. By induction of invariant 43 we have $b : \exists r_2.B \in SNF$. Due to the non-applicability of **AR8b** we have $a : \exists r.B \in SNF$.

(c) If X is of form $fc(\{b\})$, then we have $\{a\} \sqsubseteq \exists r_1.fc(\{b\}) \in SI$ and $fc(\{b\}) \sqsubseteq \exists r_2.B \in SI$, which are both contradictory to the possible form of axioms in SI. Hence we omit this situation.

Thus the induction of invariant 43 is proved.

Invariant 44: $\{a\}\exists r.\{b\} \in SI$ can be derived by **R3, 7, 8, 19**. We make a case distinction as follows:

1. If it is derived by **R3**, then there is concept X s.t. $\{a\} \sqsubseteq X, X \sqsubseteq \exists r.\{b\} \in SI$. Given the possible form of axioms in SI, X can only be a nominal $\{c\}$. Thus by induction of invariant 39 we have $a \doteq c \in SNF$. By induction of invariant 44 we have $(c, b) : r \in SNF$. Due to the non-applicability of **AR3c** we have $(a, b) : r \in SNF$.
2. If it is derived by **R7**, then there is $s \in \mathcal{RN}_\mathcal{T}$ s.t. $\{a\} \sqsubseteq \exists s.\{b\}, s \sqsubseteq r \in SI$. By induction of invariant 44 we have $(a, b) : s \in SNF$. Due to the non-applicability of **AR7b** we have $(a, b) : r \in SNF$.
3. If it is derived by **R8**, then there are a concept X and roles $r_1, r_2 \in \mathcal{RN}_\mathcal{T}$ s.t. $\{a\} \sqsubseteq \exists r_1.X, X \sqsubseteq \exists r_2.\{b\}, r_1 \circ r_2 \sqsubseteq r \in SI$. Given the possible forms of axioms in SI, X can only be a nominal $\{c\}$. By induction of invariant 44 we have $(a, c) : r_1, (c, b) : r_2 \in SNF$. Due to the non-applicability of **AR8c** we have $(a, b) : r \in SNF$.
4. If it is derived by **R19**, then we have $\{b\} \sqsubseteq \exists fi(r).\{a\} \in SI$. By induction of invariant 44 we have $(b, a) : fi(r) \in SNF$. Due to the non-applicability of **AR19** we have $(a, b) : r$.

Thus the induction of invariant 44 is proved.

With inductions of all invariants proved for each of the corresponding rules, we proved Lemma 13.

This further proves the → directions of the Theorem and hence the Theorem 12. □

Theorem 14 *For a nominal-free ontology $\mathcal{O}$, let $a, b \in \mathcal{IN}_\mathcal{O}$ be two individuals, $A, B \in \mathcal{CN}_\mathcal{O}$ be two concepts, and $r \in \mathcal{RN}_\mathcal{O}$ be a role, if for any $r_1 \circ \ldots r_n \sqsubseteq s \in \mathcal{O}$, we have $n = 2$ and $r_1 = \cdots = r_n = s$, then the following holds:*

1. $A \sqsubseteq B \in SNF_{fa}(\mathcal{O})$ iff $A \sqsubseteq B \in STNF_{fa}(\mathcal{O})$;
2. $\{a\} \sqsubseteq A \in SNF_{fa}(\mathcal{O})$ iff $a : A \in STNF_{fa}(\mathcal{O})$;
3. $\{a\} \sqsubseteq \exists r.\{b\} \in SNF_{fa}(\mathcal{O})$ iff $(a, b) : r \in STNF_{fa}(\mathcal{O})$;

Proof Similar as in the proof of Theorem 12, we use SNF to denote $SNF_{fa}(\mathcal{O})$ and use $STNF$ to denote $STNF_{fa}(\mathcal{O})$.

The first claim of the theorem is quite obvious because:

1. Both SNF and $STNF$ include $ST((\mathcal{T}, CT, QT, IT))$,
2. Neither the **AR** rules nor the **TR** rules infer any new axioms that do not involve any individual.

Thus we have $A \sqsubseteq B \in SNF$ *iff* $A \sqsubseteq B \in ST((\mathcal{T}, CT, QT, IT))$, which holds *iff* $A \sqsubseteq B \in STNF$. For the same reason, we have $A \sqsubseteq \exists r.B \in SNF$ *iff* $A \sqsubseteq \exists r.B \in STNF$.

The third claim of the theorem is also obvious because:

1. Both SNF and $STNF$ contain the same original ABox axioms from $\mathcal{A}$.
2. All entailments of form $(x, y) : r$ are either original or derived by rules **AR3c**, **7b**, **8c** and **19**.

3. All of the above rules are in both **AR** rules and **TR** rules.
4. Their premises are of forms $(x, y) : r$, $x \doteq y$, $r \sqsubseteq s$, and $r_1 \circ r_2 \sqsubseteq s$.
5. No axiom of the any of above forms is derived by the different rules between **AR** and **TR**.

Thus we know that replacing the **AR** rules with the **TR** rules will not affect the derivation of role assertion axioms. Therefore any $(x, y) : r \in SNF$, no matter being original or derived, should also be in $STNF$.

Similarly, we can also show that $x \doteq y \in SNF$ *iff* $x \doteq y \in STNF$ because all rules that derive it or any of their premises are included in both **AR** and **R**.

The ← direction of the second claim is also obvious. The rule **TR4a** is a combination of the rules **AR3a** and **AR4a**. The rule **TR4b** is a combination of the rules **AR3**, **AR4a**, **AR7a** and **AR8b**. Thus any entailment derived by any of these two rules can be derived by the **AR** rules.

The → direction of the second claim can be proved by showing that the following invariants hold:

$$a : B \in SNF \text{ only if } a : B \in STNF \tag{109}$$

$$a \not\doteq b \in SNF \text{ only if } a \not\doteq b \in STNF \tag{110}$$

$$\begin{aligned} a : \exists r.B \in SNF \text{ only if } & a : A, A \sqsubseteq \exists r.B \in STNF \\ \text{or } & (a, b) : r, b : B \in STNF \\ \text{or } & (a, b) : s, b : A, A \sqsubseteq \exists s.B, s \circ s \sqsubseteq s, s \sqsubseteq r \in STNF \end{aligned} \tag{111}$$

These invariants can also be proved with inductions. Initially, before applying either **AR** rules or **TR** rules, SNF and $STNF$ contain the same axioms thus satisfy the invariants. We prove the invariants for each of the corresponding rules.

Invariant 109: $a : B \in SNF$ can be derived by **AR1a, 1b, 2, 4a, 4b, 5a, 5b, 9a, 9b, 11**. We skip the cases of **AR1a, 1b, 2, 4b, 5b, 9a, 9b, 11**, as they can all be easily proved by induction of the above invariants and the non-applicability of rules shared by **AR** and **TR**. We consider the rest of the rules:

1. If it is derived by **AR4a**, then there are A, $A' \in \mathcal{CN}_\mathcal{T}$ and $r \in \mathcal{RN}_\mathcal{T}$ s.t. $a : \exists r.A$, $A \sqsubseteq A'$, $\exists r.A' \sqsubseteq B \in SNF$. Apparently we have $A \sqsubseteq A', \exists r.A' \sqsubseteq B \in STNF$ as well. By induction of invariant 111 we have one of the following three situations:
   (a) There is $X \in \mathcal{CN}_\mathcal{T}$ s.t. $a : X$, $X \sqsubseteq \exists r.A \in STNF$. Then we have $X \sqsubseteq B \in STNF$ due to the non-applicability of **R4**. Eventually, due to the non-applicability of **AR1a** we have $a : B \in STNF$.
   (b) There is $b \in \mathcal{IN}_\mathcal{A}$ s.t. $(a, b) : r, b : A \in STNF$. Due to the non-applicability of **AR1a** we have $b : A' \in STNF$. Due to the non-applicability of **TR4a** we have $a : B \in STNF$.
   (c) There are $C \in \mathcal{CN}_\mathcal{T}, b \in \mathcal{IN}_\mathcal{A}, s \in \mathcal{RN}_\mathcal{T}$ s.t. $(a, b) : s, b : C, C \sqsubseteq \exists s.A, s \circ s \sqsubseteq s$, $s \sqsubseteq r \in STNF$. Due to the non-applicability of **TR4b** we have $a : B \in STNF$.
2. If it is derived by **AR5a**, then $B = \bot$, and there are $A \in \mathcal{CN}_\mathcal{T}$ and $r \in \mathcal{RN}_\mathcal{T}$ s.t. $a : \exists r.A$, $A \sqsubseteq \bot \in SNF$. Apparently we have $A \sqsubseteq \bot \in STNF$ as well. By induction of invariant 111 we have one of the following three situations:
   (a) There is $X \in \mathcal{CN}_\mathcal{T}$ s.t. $a : X$, $X \sqsubseteq \exists r.A \in STNF$. Due to the non-applicability of **R5** we have $X \sqsubseteq \bot \in STNF$. Eventually, due to the non-applicability of **AR1a** we have $a : B \in STNF$.

(b) There is $b \in \mathcal{IN_A}$ s.t. $(a,b) : r, b : A \in STNF$. Due to the non-applicability of **AR1a** we have $b : \bot \in STNF$. Due to the non-applicability of **AR5b** we have $a : B \in STNF$.

(c) There are $C \in \mathcal{CN_T}, b \in \mathcal{IN_A}, s \in \mathcal{RN_T}$ s.t. $(a,b) : s, b : C, C \sqsubseteq \exists s.A, s \circ s \sqsubseteq s, s \sqsubseteq r \in STNF$. Due to the non-applicability of **R5** we will have $C \sqsubseteq \bot \in STNF$. By induction of invariant 109 we have $b : C \in STNF$. Due to the non-applicability of **AR1a** we will have $b : \bot \in STNF$. Then same as the above situation, we have $a : B \in STNF$.

Thus the induction of invariant 109 is proved.

Invariant 110: $a \neq b \in SNF$ can be derived by **AR1c, 1d, 10, 12**. The proof of all these situations is quite straightforward, simply making directly use of the induction of the above invariants.

Invariant 111: $a : \exists r.B \in SNF$ can be derived by **AR3a, 3b, 7a, 8a, 8b**. We make a case distinction:

1. If it is derived by **AR3a** then there is $A \in \mathcal{CN_T}$ s.t. $x : A, A \sqsubseteq \exists r.B \in SNF$. The later is also in $STNF$. By induction of invariant 109 we have $x : A \in STNF$. Together they satisfy the first possibility of the invariant.
2. If it is derived by **AR3b** then there is $b \in \mathcal{IN_A}$ s.t. $a \doteq b, b : \exists r.B \in SNF$. By induction of invariant 110 we have $a \doteq b \in STNF$. The later indicates one of the following three situations by induction of invariant 111:
   (a) There is $X \in \mathcal{CN_T}$ s.t. $b : X, X \sqsubseteq \exists r.B \in STNF$. Due to the non-applicability of **AR10** and **AR1b** we have $a : X \in STNF$. Together they satisfy the first possibility of the invariant.
   (b) There is $c \in \mathcal{IN_A}$ s.t. $(b,c) : r, c : B \in STNF$. Thus due to the non-applicability of **AR3c** we have $(a,c) : r \in STNF$. Together they satisfy the second possibility of the invariant.
   (c) There are $C \in \mathcal{CN_T}$, $c \in \mathcal{IN_A}$, $s \in \mathcal{RN_T}$ s.t. $(b,c) : s, c : C, C \sqsubseteq \exists s.B, s \circ s \sqsubseteq s, s \sqsubseteq r \in STNF$. Similar as above we have $(a,c) : s \in STNF$. Together they satisfy the third possibility of the invariant.
3. If it is derived by **AR7a** then there is $s \in \mathcal{RN_T}$ s.t. $a : \exists s.B, s \sqsubseteq r \in SNF$. Apparently we have $s \sqsubseteq r \in STNF$. The former is in one of the following three situations:
   (a) There is $X \in \mathcal{CN_T}$ s.t. $a : X, X \sqsubseteq \exists s.B \in STNF$. Due to the non-applicability of **R7** we have $X \sqsubseteq \exists r.B \in STNF$. Together they satisfy the first possibility of the invariant.
   (b) There is $b \in \mathcal{IN_A}$ s.t. $(a,b) : s, b : B \in STNF$. Thus due to the non-applicability of **AR7b** we have $(a,b) : r \in STNF$. Together they satisfy the second possibility of the invariant.
   (c) There are $C \in \mathcal{CN_T}, b \in \mathcal{IN_A}, t \in \mathcal{RN_T}$ s.t. $(a,b) : t, b : C, C \sqsubseteq \exists t.B, t \circ t \sqsubseteq t, t \sqsubseteq s \in STNF$. Due to the non-applicability of **R14** we also have $t \sqsubseteq r \in STNF$. Together they satisfy the third possibility of the invariant.
4. If it is derived by **AR8a** then there is $A \in \mathcal{CN_T}$ s.t. $a : \exists r.A, A \sqsubseteq \exists r.B, r \circ r \sqsubseteq r \in SNF$. Apparently we have the later two in $STNF$. The former is in one of the following three situations:
   (a) There is $X \in \mathcal{CN_T}$ s.t. $a : X, X \sqsubseteq \exists r.A \in STNF$. Due to the non-applicability of **R8** we have $X \sqsubseteq \exists r.B \in STNF$. Together they satisfy the first possibility of the invariant.

(b) There is $b \in \mathcal{IN}_\mathcal{A}$ s.t. $(a, b) : r$, $b : A \in STNF$. Together they satisfy the third possibility of the invariant, as a special case where $r = s$.

(c) There are $C \in \mathcal{CN}_\mathcal{T}$, $b \in \mathcal{IN}_\mathcal{A}$, $s \in \mathcal{RN}_\mathcal{T}$ s.t. $(a, b) : s, b : C, C \sqsubseteq \exists s.A$, $s \circ s \sqsubseteq s$, $s \sqsubseteq r \in STNF$. Due to the non-applicability of **R8** we have $C \sqsubseteq \exists s.B \in STNF$. Together they satisfy the third possibility of the invariant.

5. If it is derived by **AR8b** then there is $b \in \mathcal{IN}_\mathcal{A}$ s.t. $(a, b) : r$, $b : \exists r.B$, $r \circ r \sqsubseteq r \in SNF$. Apparently we have $(a, b) : r$, $r \circ r \sqsubseteq r \in STNF$. The $b : \exists r.B$ is in one of the following three situations:

   (a) There is $X \in \mathcal{CN}_\mathcal{T}$ s.t. $b : X$, $X \sqsubseteq \exists r.B \in STNF$. Together they satisfy the third possibility of the invariant, as a special case where $r = s$.

   (b) There is $c \in \mathcal{IN}_\mathcal{A}$ s.t. $(b, c) : r$, $c : B \in STNF$. Thus due to the non-applicability of **AR8c** we have $(a, c) : r \in STNF$. Together they satisfy the third possibility of the invariant, as a special case where $r = s$.

   (c) There are $C \in \mathcal{CN}_\mathcal{T}$, $c \in \mathcal{IN}_\mathcal{A}$, $s \in \mathcal{RN}_\mathcal{T}$ s.t. $(b, c) : s, c : C, C \sqsubseteq \exists s.B$, $s \circ s \sqsubseteq s$, $s \sqsubseteq r \in STNF$. Due to the non-applicability of **AR7b** we have $(b, c) : r \in STNF$. Due to the non-applicability of **AR8c** we have $(a, c) : r \in STNF$. Due to the non-applicability of **R7** we have $C \sqsubseteq \exists r.B \in STNF$. Together they satisfy the third possibility of the invariant.

Thus the induction of invariant 111 is proved. Thus we prove the → direction of the second claim of the theorem and hence prove the theorem. □

Theorem 15 *For a nominal-safe ontology $\mathcal{O}$ under approximate reasoning, let $a, b \in \mathcal{IN}_\mathcal{O}$ be two individuals, $A, B \in \mathcal{CN}_\mathcal{O}$ be two concepts, and $r \in \mathcal{RN}_\mathcal{O}$ be a role, then the following holds:*

1. $A \sqsubseteq B \in SI_{fa}(\mathcal{O})$ iff $A \sqsubseteq B \in SC_{fa}(\mathcal{O})$;
2. $\{a\} \sqsubseteq A \in SI_{fa}(\mathcal{O})$ iff $a : A \in SC_{fa}(\mathcal{O})$;
3. $\{a\} \sqsubseteq \exists r.\{b\} \in SI_{fa}(\mathcal{O})$ iff $(a, b) : r \in SC_{fa}(\mathcal{O})$;

Proof The theorem is similar to Theorem 12 hence we follow a similar proof.

The ← direction is obvious because the additional **CR** rules are special cases of the corresponding **R** rules where certain concepts are restricted to nominals. If we rewrite the ABox axioms into corresponding nominal form TBox axioms, the **CR** rules are the same as or can be reduced to combinations of **R** rules.

Let $A_{fa}(\mathcal{O}) = (\mathcal{T}, \mathcal{A}, CT, QT, IT)$ and $AI(A_{fa}(\mathcal{O})) = \{\mathcal{T}', CT, QT, IT\}$. In the following, we use SI to denote $SI_{fa}(\mathcal{O})$, and SC to denote $SC_{fa}(\mathcal{O})$.

Before we prove the → direction, we analyse the forms of axioms in SI. Because the ontology is nominal-safe, all axioms in the initial $\mathcal{T}'$ should not have $fc(\{a\})$ as the LHS if its RHS is not of the same form, or $\{a\}$ as the RHS if its LHS is not a nominal.

Hence, as a consequence of Theorem 6, we know that for a nominal-safe ontology $\mathcal{O}$, SI have the following properties, where A, B, $fc(A)$, $fc(B) \in \mathcal{CN}_\mathcal{T}$ and a, b are individuals:

1. $A \sqsubseteq \{a\} \notin SI$ unless $A \sqsubseteq \bot \in SI$;
2. $fc(\{a\}) \sqsubseteq B \notin SI$ unless $\top \sqsubseteq B \in SI$;
3. $fc(\{a\}) \sqsubseteq \{b\}$;
4. $fc(\{a\}) \sqsubseteq \exists r.A \notin SI$ unless $\top \sqsubseteq \exists r.A \in SI$;
5. $fc(\{a\}) \sqsubseteq \exists r.\{b\} \notin SI$ unless $\top \sqsubseteq \exists r.\{b\} \in SI$;

6. $fc(\{a\}) \sqsubseteq \exists r.fc(\{b\}) \notin SI$ unless $\top \sqsubseteq \exists r.fc(\{b\}) \in SI$;
7. $C_1 \sqcap \cdots \sqcap C_n \sqsubseteq \{a\} \notin SI$ where C_i is a named concept, nominal, $\top$ or $\bot$;

Furthermore, because the **R** rules do not infer axioms of form $C_1 \sqcap \cdots \sqcap C_n \sqsubseteq C_{n+1}$ or $\exists r.C_1 \sqsubseteq C_2$, we know that for any axiom α of one of these two forms, we have $\alpha \in SI$ *iff* $\alpha \in SC$.

Thus we only need to consider the other kind of entailments in SI.

We prove the → directions of the theorem by proving the following stronger lemma:

Lemma 16 *For an ontology $\mathcal{O}$, let $A_{fa}(\mathcal{O}) = (\mathcal{T}, \mathcal{A}, CT, QT, IT)$ and $AI(A_{fa}(\mathcal{O}))$ be $(\mathcal{T}', CT, QT, IT)$. If we use SI to denote $SI_{fa}(\mathcal{O})$, and SC to denote $SC_{fa}(\mathcal{O})$, then the following invariants hold in each step of computation of SI, where $a, b \in \mathcal{IN}_{\mathcal{O}}$ are two individuals, $A, B \in \mathcal{CN}_{\mathcal{T}}$ are two concepts, and $r \in \mathcal{RN}_{\mathcal{O}}$ is a role:*

$$fc(A), fc(B) \in \mathcal{CN}_{\mathcal{T}} \textit{ and } A \sqsubseteq B \in SI \textit{ only if } A \sqsubseteq B \in SC \tag{57}$$

$$fc(A) \in \mathcal{CN}_{\mathcal{T}} \textit{ and } A \sqsubseteq fc(\{a\}) \in SI \textit{ only if } a : fc(A) \in SC \tag{58}$$

$$A \sqsubseteq \bot \notin SI \textit{ and } A \sqsubseteq \exists r.B \in SI \textit{ only if } A \sqsubseteq \exists r.B \in SC \tag{59}$$

$$A \sqsubseteq \exists r.\{a\} \textit{ only if } A \sqsubseteq \exists r.\{a\} \in SC \tag{60}$$

$$\{a\} \sqsubseteq B \in SI \textit{ only if } a : B \in SC \tag{61}$$

$$\{a\} \sqsubseteq \{b\} \in SI \textit{ only if } a \doteq b \in SC \tag{62}$$

$$fc(\{a\}) \sqsubseteq fc(\{b\}) \in SI \textit{ only if } a \doteq b \in SC \tag{63}$$

$$r \sqsubseteq s \in SI \textit{ only if } r \sqsubseteq s \in SC \tag{64}$$

$$\{a\} \sqsubseteq fc(\{b\}) \in SI \textit{ only if } a \not\doteq b \in SC \tag{65}$$

$$\{a\} \sqsubseteq \exists r.\{b\} \in SI \textit{ only if } (a, b) : r \in SC \tag{66}$$

$$r_1 \circ r_2 \sqsubseteq s \in SI \textit{ only if } r_1 \circ r_2 \sqsubseteq s \in SC \tag{67}$$

$$\{a\} \sqsubseteq \bot \notin SI \textit{ and } \{a\} \sqsubseteq \exists r.B \in SI \textit{ only if } a : \exists r.B \in SC \tag{68}$$

Apparently the → directions of the theorem hold if the above lemma holds.

To prove the above invariants, we perform induction of the application of rules. First of all, it is obvious that the invariants are satisfied by all initial elements of SI, i.e. all axioms in $\mathcal{T}'$ and corresponding tautologies added into $SI = ST((\mathcal{T}', CT, QT, IT))$ in Step-1 of Def. 5.

Then we show that, the above invariants hold for each of the corresponding axioms derived by each of the corresponding rules.

Invariant 57: $A \sqsubseteq B \in SI$ can be derived by **R1, 2, 4, 5, 9, 10, 11, 12, 13**. Note that **R6** can not derive $A \sqsubseteq B \in SI$ when the $\mathcal{O}$ is nominal-safe. We make a case distinction as follows:

1. If it is derived by **R1**. Then there is a concept X s.t. $A \sqsubseteq X$ and $X \sqsubseteq B$. We make a case distinction on the form of X:
   (a) If $X, fc(X) \in \mathcal{CN}_{\mathcal{T}}$, by induction of invariant 57 we know that $A \sqsubseteq X$, $X \sqsubseteq B \in SC$. Thus by non-applicability of rule **R1** we have $A \sqsubseteq B \in SC$.
   (b) If X is a nominal $\{a\}$, then we have $A \sqsubseteq \{a\} \in SI$, which is contradictory to the form of axioms appearing in SI unless $A \sqsubseteq \bot \in SI$. By induction of invariant 57 we have $A \sqsubseteq \bot \in SC$. Together with $\bot \sqsubseteq B \in SC$ we directly have $A \sqsubseteq B \in SC$.
   (c) If X is of form $fc(\{b\})$, then we have $fc(\{b\}) \sqsubseteq B \in SI$, which is contradictory to the form of axioms appearing in SI unless $\top \sqsubseteq B \in SI$. By induction of invariant 57 we have $\top \sqsubseteq B \in SC$. Together with $A \sqsubseteq \top \in SC$ we directly have $A \sqsubseteq B \in SC$.

2. If it is derived by **R2**, then there are concepts $A_1, \ldots, A_n$ s.t. $A \sqsubseteq A_i \in SI$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq B \in \mathcal{T}'$. Given the form of the later, it should also be in $\mathcal{T}$. We make a case distinction on the form of A_i as follows:
   (a) If $A_i, fc(A_i) \in \mathcal{CN}_{\mathcal{T}}$, then by induction of invariant 57 we have $A \sqsubseteq A_i \in SC$.
   (b) If A_i is a nominal $\{a\}$, then we have $A \sqsubseteq \{a\} \in SI$, which is contradictory to the form of axioms appearing in SI unless $A \sqsubseteq \bot \in SI$, in which case by induction of invariant 57 we have $A \sqsubseteq \bot \in SC$ and hence $A \sqsubseteq B \in SC$.
   (c) If A_i is of form $fc(\{a\})$, they by induction of invariant 58 we have $a : fc(A) \in SC$. Given that $fc(\{a\})$ is in $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq B$, $\{a\}$ is native and introduced. Thus due to the non-applicability of **IR1** we have $\{a\} \sqsubseteq fc(A) \in SC$. Due to the non-applicability of **R10** we have $A \sqsubseteq fc(\{a\}) \in SC$.

   Together, due to the non-applicability of **R2** we have $A \sqsubseteq B \in SC$.
3. If it is derived by **R4**, then there is $r \in \mathcal{RN}_{\mathcal{T}'}$ and two concepts X and X' such that $A \sqsubseteq \exists r.X \in SI$, $X \sqsubseteq X' \in SI$ and $\exists r.X' \sqsubseteq B \in \mathcal{T}'$. Given the form of $\exists r.X' \sqsubseteq B$, it should also be in $\mathcal{T}$. We make a case distinction on the form of X and X':
   (a) If $X, fc(X) \in \mathcal{CN}_{\mathcal{T}}$, when $A \sqsubseteq \bot \in SI$, then by induction of invariant 57 we have $A \sqsubseteq \bot \in SC$ and hence $A \sqsubseteq B \in SC$. Otherwise, by induction of invariant 59 we know that $A \sqsubseteq \exists r.X \in SC$:
      i. If $X', fc(X') \in \mathcal{CN}_{\mathcal{T}}$, by induction of invariant 57 we also know that $X \sqsubseteq X' \in SC$. Thus by non-applicability of rule **R4** we have $A \sqsubseteq B \in SC$.
      ii. If X' is a nominal $\{a\}$, then we have $X \sqsubseteq \{a\} \in SI$, which is contradictory to the possible forms of axioms in SI unless $X \sqsubseteq \bot \in SI$. In this case, we will have $X \sqsubseteq \bot \in SC$ by induction of invariant 57. Together due to the non-applicability of **R5** we also have $A \sqsubseteq \bot \in SC$ and hence $A \sqsubseteq B \in SC$.
      iii. If X' is of form $fc(\{a\})$, then by induction of invariant 58 we have $a : fc(X) \in SC$. Because $\exists r.fc(\{a\}) \sqsubseteq B \in T'$, $\{a\}$ is native. Hence due to the non-applicability of **IR1** and **R10** we have $X \sqsubseteq fc(\{a\}) \in SC$. Due to the non-applicability of **R4** we have $A \sqsubseteq B \in SC$.
   (b) If X is a nominal $\{a\}$, then we have $A \sqsubseteq \exists r.\{a\} \in SI$. By induction of invariant 60 we have $A \sqsubseteq \exists r.\{a\} \in SC$. This further indicates that $\{a\}$ has already been introduced:
      i. If $X', fc(X') \in \mathcal{CN}_{\mathcal{T}}$, by induction of invariant 61 we have $a : X' \in SC$. Given that $\{a\}$ is introduced, due to the non-applicability of **IR1** we have $\{a\} \sqsubseteq X' \in SC$.
      ii. If X' is another nominal $\{b\}$, then we have $\{a\} \sqsubseteq \{b\} \in SI$. By induction of invariant 62 we have $a \doteq b \in SC$. Because $\exists r.\{b\} \sqsubseteq B \in \mathcal{T}$, $\{b\}$ is also introduced. Hence due to the non-applicability of **IR3** we have $\{a\} \sqsubseteq \{b\} \in SC$.
      iii. If X' is of form $fc(\{b\})$, it is similar as the above situation that eventually we have $\{a\} \sqsubseteq fc(\{b\}) \in SC$.

      In any case, due to the non-applicability of **R4** we have $A \sqsubseteq B \in SC$.
   (c) If X is of form $fc(\{a\})$, then we have $A \sqsubseteq \exists r.fc(\{a\}) \in SI$. By induction of invariant 59 we have $A \sqsubseteq \exists r.fc(\{a\}) \in SC$. This further indicates that $fc(\{a\})$ is native:
      i. If $X', fc(X') \in \mathcal{CN}_{\mathcal{T}}$, then we have $fc(\{a\}) \sqsubseteq X' \in SI$, which is contradictory to the possible forms of axioms in SI unless we have $\top \sqsubseteq X' \in SI$.

By induction of invariant 57 we have $\top \sqsubseteq X' \in SC$. Because $fc(\{a\})$ is native, we have $fc(\{a\}) \sqsubseteq \top \in SC$. Due to the non-applicability of **R1** we have $fc(\{a\}) \sqsubseteq X' \in SC$. Due to the non-applicability of **R4** we have $A \sqsubseteq B \in SC$.

ii. If X' is another nominal $\{b\}$, then we have $fc(\{a\}) \sqsubseteq \{b\} \in SI$, which is contradictory to the possible forms of axioms in SI thus we omit this situation.

iii. If X' is of form $fc(\{b\})$, then we have $fc(\{a\}) \sqsubseteq fc(\{b\}) \in SI$. By induction of invariant 63 we have $a \doteq b \in SC$. Because $\exists r.fc(\{b\}) \sqsubseteq B \in \mathcal{T}$ we know that $fc(\{b\})$ is native. Due to non-applicability of **IR3** and **R10** we have $fc(\{a\}) \sqsubseteq fc(\{b\}) \in SC$. Due to non-applicability of **R4** we have $A \sqsubseteq B \in SC$.

4. If it is derived by **R5**, then $B = \bot$ and there is concept X and $r \in \mathcal{RN}_\mathcal{T}$ s.t. $A \sqsubseteq \exists r.X \in SI$ and $X \sqsubseteq \bot \in SI$. We make a case distinction on the form of X:
   (a) If $X, fc(X) \in \mathcal{CN}_\mathcal{T}$. When $A \sqsubseteq \bot \in SI$ we trivially have $A \sqsubseteq B \in SC$ as before. Otherwise by invariant 59 we have $A \sqsubseteq \exists r.X \in SC$ and $X \sqsubseteq \bot \in SC$. Due to the non-applicability of **R5** we have $A \sqsubseteq B \in SC$.
   (b) If X is a nominal $\{a\}$, then we have $A \sqsubseteq \exists r.\{a\} \in SI$. By induction of invariant 60 we have $A \sqsubseteq \exists r.\{a\} \in SC$. This also indicates that $\{a\}$ is introduced. We also have $\{a\} \sqsubseteq \bot \in SI$. By induction of invariant 61 we have $a : \bot \in SC$. Due to the non-applicability of **IR1** we have $\{a\} \sqsubseteq \bot \in SC$. Together due to the non-applicability of **R5** we have $A \sqsubseteq \bot \in SC$.
   (c) If X is of form $fc(\{a\})$, then we have $fc(\{a\}) \sqsubseteq \bot \in SI$, which is contradictory to the possible forms of axioms in SI thus we omit this situation.
5. If it is derived by **R9**, then $B = \bot$ and there is a concept X s.t. $A \sqsubseteq X$ and $A \sqsubseteq fc(X)$. We make a case distinction on form of X:
   (a) If X, $fc(X) \in \mathcal{CN}_\mathcal{T}$, thus by invariant 57 we have $A \sqsubseteq X, A \sqsubseteq fc(X) \in SC$. Due to the non-applicability of **R9** we have $A \sqsubseteq \bot \in SC$.
   (b) If X is a nominal $\{a\}$, then we have $A \sqsubseteq \{a\} \in SI$. Given the possible form of axioms in SI, we have $A \sqsubseteq \bot \in SI$. By induction of invariant 57 we directly have $A \sqsubseteq B \in SC$.
   (c) If X is of form $fc(\{a\})$, then we have $A \sqsubseteq fc(fc(\{a\})) \in SI$, i.e. $A \sqsubseteq \{a\} \in SI$. Then we skip this situation as it is the same as above.
6. If it is derived by **R10**, then we have $fc(B) \sqsubseteq fc(A) \in SI$. We also have $fc(A)$, $fc(B) \in \mathcal{CN}_\mathcal{T}$, by induction on invariant 57 we have $fc(B) \sqsubseteq fc(A) \in SC$. Due to the non-applicability of rule **R10** we have $A \sqsubseteq B \in SC$.
7. If it is derived by **R11**, then we have $A_1, \ldots, A_n$ s.t. $B = fc(A_i)$ and $A \sqsubseteq A_j \in SI(1 \leq j \leq n, j \neq i)$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq \bot \in \mathcal{T}$. We make a case distinction on the form of A_j as follows:
   (a) If $A_j, fc(A_j) \in \mathcal{CN}_\mathcal{T}$, then by induction of invariant 57 we have $A \sqsubseteq A_j \in SC$.
   (b) If A_j is a nominal $\{a\}$, then we have $A \sqsubseteq \{a\} \in SI$, which is contradictory to the form of axioms appearing in SI unless $A \sqsubseteq \bot \in SI$, in which case by induction of invariant 57 we have $A \sqsubseteq \bot \in SC$ and hence $A \sqsubseteq B \in SC$.
   (c) If A_j is of form $fc(\{a\})$, they by induction of invariant 58 we have $a : fc(A) \in SC$. Given that $fc(\{a\})$ is in $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq \bot \in \mathcal{T}$, $\{a\}$ is native. Thus due to the non-applicability of **IR1** we have $\{a\} \sqsubseteq fc(A) \in SC$. Due to the non-applicability of **R10** we have $A \sqsubseteq fc(\{a\}) \in SC$.

Together, due to the non-applicability of **R11** we have $A \sqsubseteq B \in SC$.

8. If it is derived by **R12**, then there is individual $a \in \mathcal{IN}_\mathcal{A}$ and role $r \in \mathcal{RN}_\mathcal{T}$ s.t. $A \sqsubseteq \exists r.\{a\} \in SI$ and $fc(B) \sqsubseteq \exists fc(r).\{a\} \in SI$. By induction of invariant 60 we have $A \sqsubseteq \exists r.\{a\} \in SC$ and $fc(B) \sqsubseteq \exists fc(r).\{a\} \in SC$. Due to the non-applicability of **R12** we have $A \sqsubseteq B \in SC$.
9. If it is derived by **R13**, then there are concepts X and Y, roles r, $s \in \mathcal{RN}_\mathcal{T}$ and integers i and j s.t. $X \sqsubseteq Y \in SI$, $(A, X, r, i) \in QT$, $(B, Y, s, j) \in QT$, $r \sqsubseteq s \in SI$ and $i \geq j$. By induction of invariant 64 we have $r \sqsubseteq s \in SC$. $(A, X, r, i), (B, Y, s, j) \in QT$ should also remain as QT is the same in computation of SI and SC. And it's apparent that $i \geq j$ remains the same. We make a case distinction on the form of X and Y as follows:
   (a) If $X, fc(X) \in \mathcal{CN}_\mathcal{T}$:
      i. If $Y, fc(Y) \in \mathcal{CN}_\mathcal{T}$, then by induction of invariant 57 we have $X \sqsubseteq Y \in SC$.
      ii. If Y is a nominal $\{a\}$, then we have $X \sqsubseteq \{a\}$, which is contradictory to the possible forms of axioms in SI unless $X \sqsubseteq \bot \in SI$. In the later case, by induction of invariant 57 we also have $X \sqsubseteq \bot \in SC$. According to step-6 of Def. 4 we have $A \sqsubseteq \exists r.X \in \mathcal{T} \subseteq SC$. Due to non-applicability of **R5** we have $A \sqsubseteq \bot \in SC$ hence $A \sqsubseteq B \in SC$.
      iii. If Y is of form $fc(\{a\})$, then by induction of invariant 58 we have $a : fc(X) \in SC$. Meanwhile, $(B, fc(\{a\}), s, j) \in QT$ indicates that $fc(\{a\})$ is native. Hence due to non-applicability of **IR1** and **R10** we still have $X \sqsubseteq fc(\{a\}) = X \sqsubseteq Y \in SC$.
   (b) If X is a nominal $\{a\}$, because $(A, \{a\}, r, i) \in QT$ we know that $\{a\}$ is native:
      i. If $Y, fc(Y) \in \mathcal{CN}_\mathcal{T}$, then by induction of invariant 61 we have $a : Y \in SC$. Due to the non-applicability of **IR1** we have $X \sqsubseteq Y = \{a\} \sqsubseteq Y \in SC$.
      ii. If Y is also a nominal $\{b\}$, then by induction of invariant 62 we have $a \doteq b \in SC$. We also know that $\{b\}$ is a native nominal. Due to the non-applicability of **IR3** we have $X \sqsubseteq Y = \{a\} \sqsubseteq \{b\} \in SC$.
      iii. If Y is of form $fc(\{b\})$, then by induction of invariant 65 we have $a \not\doteq b \in SC$. We also know that $fc(\{b\})$ is native. Due to the non-applicability of **IR4** we have $X \sqsubseteq Y = \{a\} \sqsubseteq fc(\{b\}) \in SC$.
   (c) If X is of form $fc(\{a\})$, we know that $fc(\{a\})$ is native:
      i. If $Y, fc(Y) \in \mathcal{CN}_\mathcal{T}$, then we have $fc(\{a\}) \sqsubseteq Y \in SI$, which is contradictory to the possible forms of axioms in SI unless $\top \sqsubseteq Y \in SI$. In the later case, by induction of invariant 57 we have $\top \sqsubseteq Y \in SC$. Because we also have $fc(\{a\}) \sqsubseteq \top \in SC$ we can derive $fc(\{a\}) \sqsubseteq Y \in SC$.
      ii. If Y is also a nominal $\{b\}$, then we have $fc(\{a\}) \sqsubseteq \{b\} \in SI$, which is contradictory to the possible forms of axioms in SI.
      iii. If Y is of form $fc(\{b\})$, by induction of invariant 63 we have $a \doteq b \in SC$ and hence $b \doteq a \in SC$. Because $\{b\}$ is also a native nominal, due to non-applicability of **IR3** we have $\{b\} \sqsubseteq \{a\} \in SC$. Due to non-applicability of **R10** we have $X \sqsubseteq Y = fc(\{a\}) \sqsubseteq fc(\{b\}) \in SC$.

   In all the above situations where X is satisfiable, we have $X \sqsubseteq Y \in SC$. Hence due to the non-applicability of **R13** we have $A \sqsubseteq B \in SC$.

Thus the induction of invariant 57 is proved.

Invariant 58: $A \sqsubseteq fc(\{a\}) \in SI$ can be derived by **R1, 2, 4, 10, 11, 12**. We skip the obvious situation where $A = \bot$ and make a case distinction as follows:

1. If it is derived by **R1**, then there is a concept X s.t. $A \sqsubseteq X \in SI$ and $X \sqsubseteq fc(\{a\}) \in SI$. We make a case distinction on the form of X:
   (a) If $X, fc(X) \in \mathcal{CN}_{\mathcal{T}}$, then by induction of invariant 57 we have $A \sqsubseteq X \in SC$. Due to the non-applicability of **R10** we have $fc(X) \sqsubseteq fc(A) \in SC$. By induction of invariant 58 we have $a : fc(X) \in SC$. Due to the non-applicability of **AR1a** we have $a : fc(A) \in SC$.
   (b) If X is a nominal $\{b\}$, then we have $A \sqsubseteq \{b\} \in SI$, which is contradictory to the possible forms of axioms in SI unless we have $A \sqsubseteq \bot \in SI$. By induction of invariant 57 we have $A \sqsubseteq \bot \in SC$. Due to the non-applicability of **R10** we also have $\top \sqsubseteq fc(A) \in SC$. Given that $a : \top \in SC$, due to the non-applicability of **AR1a** we have $a : fc(A) \in SC$.
   (c) If X is of form $fc(\{b\})$, then by induction of invariant 58, we have $b : fc(A) \in SC$. By induction of invariant 63 we have $a \doteq b \in SC$. Due to the non-applicability of **AR1b** we have $a : fc(A) \in SC$.
2. If it is derived by **R2**, then there are concepts $A_1, \ldots, A_n$ s.t. $A \sqsubseteq A_i \in SI$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq fc(\{a\}) \in \mathcal{T}$. Hence $\{a\}$ is a native nominal. In this case, $A \sqsubseteq fc(\{a\}) \in SC$ can be proved in the same way as $A \sqsubseteq B \in SC$. Due to the non-applicability of **R10** we have $\{a\} \sqsubseteq fc(A) \in SC$. Due to the non-applicability of **IR1** we have $a : fc(A) \in SC$.
3. If it is derived by **R4**, then there is $r \in \mathcal{RN}_{\mathcal{T}}$ and two concepts X and X' such that $A \sqsubseteq \exists r.X \in SI$, $X \sqsubseteq X' \in SI$ and $\exists r.X' \sqsubseteq fc(\{a\}) \in \mathcal{T}'$. Given the form of $\exists r.X' \sqsubseteq fc(\{a\})$, it should also be in $\mathcal{T}$. This indicates that $\{a\}$ is a native nominal. In this case, $A \sqsubseteq fc(\{a\}) \in SC$ can be proved in the same way as $A \sqsubseteq B \in SC$. Then similar as the above situation we have $a : fc(A) \in SC$.
4. If it is derived by **R10**, then there is $\{a\} \sqsubseteq fc(A) \in SI$. By induction of invariant 61 we have $a : fc(A) \in SC$.
5. If it is derived by **R11**, then we have $A_1, \ldots, A_n$ s.t. $A_i = \{a\}$ and $A \sqsubseteq A_j \in SI$ $(1 \leq j \leq n, j \neq i)$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq \bot \in \mathcal{T}$. Given that A_i is in an original axiom in $\mathcal{T}$, we know that $\{a\}$ is a native nominal. In this case, $A \sqsubseteq fc(\{a\}) \in SC$ can be proved in the same way as $A \sqsubseteq B \in SC$. Then similar as the **R4** situation we have $a : fc(A) \in SC$.
6. If it is derived by **R12**, then there is individual $b \in \mathcal{IN}_{\mathcal{A}}$ and role $r \in \mathcal{RN}_{\mathcal{T}}$ s.t. $A \sqsubseteq \exists r.\{b\} \in SI$ and $\{a\} \sqsubseteq \exists fc(r).\{b\} \in SI$. By induction of invariant 60 we have $A \sqsubseteq \exists r.\{b\} \in SC$. By induction of invariant 66 we have $(a, b) : fc(r) \in SC$. Due to the non-applicability of **CR12** we have $a : fc(A) \in SC$.

Thus the induction of invariant 58 is proved.

Invariant 59: $A \sqsubseteq \exists r.B \in SI$ can be derived by **R3, 7, 8**. Note that $A \sqsubseteq \bot \notin SI$. The situations where it is derived by **R3** or **R7** can be proved in the same way as invariant 42 in the proof of Theorem 12 so we skip them.

If it is derived by **R8**, there are concept X and roles $r_1, r_2 \in \mathcal{RN}_{\mathcal{T}}$ s.t. $A \sqsubseteq \exists r_1.X, X \sqsubseteq \exists r_2.B, r_1 \circ r_2 \sqsubseteq R \in SI$. By induction of invariant 67 we have $r_1 \circ r_2 \sqsubseteq R \in SC$. We make a case distinction on the form of X as follows:

1. If $X, fc(X) \in \mathcal{CN}_{\mathcal{T}}$. By induction of invariant 59 we have $A \sqsubseteq \exists r_1.X, X \sqsubseteq \exists r_2.B \in SC$. Due to the non-applicability of **R8** we have $A \sqsubseteq \exists r.B \in SC$.
2. If X is a nominal $\{a\}$, then we have $A \sqsubseteq \exists r_1.\{a\} \in SI$ and $\{a\} \sqsubseteq \exists r_2.B \in SI$. By induction of invariant 60 we have $A \sqsubseteq \exists r_1.\{a\} \in SC$, implying that $\{a\}$ is an introduced nominal. By induction of invariant 68 we have $a : \exists r_2.B \in SC$. Because $\{a\}$ is

introduced, due to the non-applicability of **IR5** we have $\{a\} \sqsubseteq \exists r_2.B \in SC$. Due to the non-applicability of **R8** we have $A \sqsubseteq \exists r.B \in SC$.

3. If X is of form $fc(\{a\})$, then we have $fc(\{a\}) \sqsubseteq \exists r_2.B \in SI$. This occurs only when we have $\top \sqsubseteq \exists r._2B \in SI$. By induction of invariant 59 we have $\top \sqsubseteq \exists r._2B \in SC$. Since we have $fc(\{a\})$ introduced, we know that it is also native. Hence we have $fc(\{a\}) \sqsubseteq \top \in SC$. Due to the non-applicability of **R3** we have $fc(\{a\}) \sqsubseteq \exists r._2B \in SC$. Together due to the non-applicability of **R8** we have $A \sqsubseteq \exists r.B \in SC$.

Thus the induction of invariant 59 is proved.

Invariant 60: $A \sqsubseteq \exists r.\{a\} \in SI$ can be derived by **R3, 7, 8**. Note that $A \sqsubseteq \bot \notin SI$. We make a case distinction as follows:

1. If it is derived by **R3**, then there is concept X s.t. $A \sqsubseteq X \in SI$ and $X \sqsubseteq \exists r.\{a\} \in SI$. Given the possible forms of axioms in SI we have $X \in \mathcal{CN}_\mathcal{T}$. By induction of invariant 57 we have $A \sqsubseteq X \in SC$. By induction of invariant 60 we have $X \sqsubseteq \exists r.\{a\} \in SC$. Due to the non-applicability of **R3** we have $A \sqsubseteq \exists r.\{a\} \in SC$.
2. If it is derived by **R7**, then simply by induction of invariant 60 and invariant 64, and due to the non-applicability of **R7** we have $A \sqsubseteq \exists r.\{a\} \in SC$.
3. If it is derived by **R8**, there are concept X and roles $r_1, r_2 \in \mathcal{RN}_\mathcal{T}$ s.t. $A \sqsubseteq \exists r_1.X, X \sqsubseteq \exists r_2.\{a\}, r_1 \circ r_2 \sqsubseteq r \in SI$. By induction of invariant 67 we have $r_1 \circ r_2 \sqsubseteq r \in SC$. We make a case distinction on the form of X as follows:
   (a) If $X, fc(X) \in \mathcal{CN}_\mathcal{T}$. By induction of invariant 59 we have $A \sqsubseteq \exists r_1.X$. By induction of invariant 60, we have $X \sqsubseteq \exists r_2.\{a\} \in SC$. Due to the non-applicability of **R8** we have $A \sqsubseteq \exists r.\{a\} \in SC$.
   (b) If X is a nominal $\{b\}$, then we have $A \sqsubseteq \exists r_1.\{b\} \in SI$, and $\{b\} \sqsubseteq \exists r_2.\{a\} \in SI$. By induction of invariant 60 we have $A \sqsubseteq \exists r_1.\{a\} \in SC$. By induction of invariant 66 we have $(b, a) : r_2 \in SC$. Due to the non-applicability of **CR8** we have $A \sqsubseteq \exists r.\{a\} \in SC$. This is also where $\{a\}$ is introduced in SC.
   (c) If X is of form $fc(\{b\})$, then we have $fc(\{b\}) \sqsubseteq \exists r_2.\{a\} \in SI$, which is contradictory to the possible forms of axioms in SI unless there is $\top \sqsubseteq \exists r_2.\{a\} \in SI$. In this case, because $fc(\{b\})$ is native, eventually we will have $fc(\{b\}) \sqsubseteq \exists r_2.\{a\} \in SC$. Due to the non-applicability of **R8** we have $A \sqsubseteq \exists r.\{a\} \in SC$.

Thus the induction of invariant 60 is proved.

Invariant 61: $\{a\} \sqsubseteq B \in SI$ can be derived by **R1, 2, 4, 5, 9, 10, 11, 12**. The situations where it is derived by **R1, 5, 9, 10** can be proved in the same way as invariant 38 in the proof of Theorem 12 so we skip them.

We make a case distinction on the other rules as follows:

1. If it is derived by **R2**, then there are concepts $A_1, \ldots, A_n$ s.t. $\{a\} \sqsubseteq A_i \in SI$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq B \in \mathcal{T}'$. Given the form of the later, it should also be in $\mathcal{T}$. We make a case distinction on the form of A_i as follows:
   (a) If $A_i, fc(A_i) \in \mathcal{CN}_\mathcal{T}$, then by induction of invariant 61 we have $a : A_i \in SC$.
   (b) If A_i is a nominal $\{b\}$, then by induction of invariant 62 we have $a \doteq b \in SC$. Given that $\{b\}$ is native and introduced, due to the non-applicability of **IR6** we have $a : \{b\} \in SC$.
   (c) If A_i is of form $fc(\{b\})$, they by induction of invariant 65 we have $a \not\doteq b \in SC$. Given that $fc(\{b\})$ is native an introduced, due to the non-applicability of **IR7** we have $a : fc(\{b\}) \in SC$.

Together, due to the non-applicability of **AR2** we have $x : B \in SNF$.

2. If it is derived by **R4**, then there is $r \in \mathcal{RN_T}$ and two concepts X and X' such that $\{a\} \sqsubseteq \exists r.X \in SI$, $X \sqsubseteq X' \in SI$ and $\exists r.X' \sqsubseteq B \in \mathcal{T}'$. Given the form of $\exists r.X' \sqsubseteq B$, it should also be in $\mathcal{T}$. We skip the obvious situation where $\{a\} \sqsubseteq \bot \in SI$ before **R4** is applied. We make a case distinction on the form of X and X':
   (a) If $X, fc(X) \in \mathcal{CN_T}$, by induction of invariant 68 we know that $a : \exists r.X \in SC$:
      i. If $X', fc(X') \in \mathcal{CN_T}$, by induction of invariant 57 we also know that $X \sqsubseteq X' \in SC$. Thus by non-applicability of rule **AR4a** we have $a : B \in SC$.
      ii. If X' is a nominal $\{b\}$, then we have $X \sqsubseteq \{b\} \in SI$, which is contradictory to the possible forms of axioms in SI except that $X \sqsubseteq \bot \in SI$. In the later case, it is apparent that we can derive $a : \bot \in SC$ hence $a : B \in SC$.
      iii. If X' is of form $fc(\{b\})$, then by induction of invariant 58 we have $b : fc(X) \in SC$. Because $\exists r.fc(\{b\}) \sqsubseteq B \in \mathcal{T}$, $\{b\}$ and $fc(\{b\})$ are native. hence due to the non-applicability of **IR1** and **R10** we have $X \sqsubseteq fc(\{b\}) \in SC$. Due to the non-applicability of **AR4a** we have $a : B \in SC$.
   (b) If x is a nominal $\{b\}$, by induction of invariant 66 we have $(a, b) : r \in SC$:
      i. If $X'.fc(X') \in \mathcal{CN_T}$, by induction of invariant 61 we have $b : X' \in SC$.
      ii. If X' is a nominal $\{c\}$, by induction of invariant 62 we have $b \doteq c \in SC$. Because $\exists r.fc(\{c\}) \in \mathcal{T}$ we know that $\{c\}$ is native and introduced. Hence due to the non-applicability of **IR6** we have $b : \{c\} \in SC$.
      iii. If X' is of form $fc(\{c\})$, then similar as the above situation, due to the non-applicability of **IR7** we have $b : fc(\{c\}) \in SC$.

      In all above situations, by non-applicability of rule **AR4b** we have $a : B \in SC$.
   (c) If X is of form $fc(\{b\})$, then we have $\{a\} \sqsubseteq \exists r.fc(\{b\}) \in SI$. By induction of invariant 68 we have $a : \exists r.fc(\{b\}) \in SC$, which further suggests that $\{b\}$ is native:
      i. If $X', fc(X') \in \mathcal{CN_T}$, then we have $fc(\{b\}) \sqsubseteq X' \in SI$, which is contradictory to the possible forms of axioms in SI unless $T \sqsubseteq X' \in SI$. Then similar as the proof of situation **R4** of invariant 57 we have $fc(\{b\}) \sqsubseteq X' \in SC$. Due to the non-applicability of **AR4a** we have $a : B \in SC$.
      ii. If X' is a nominal $\{c\}$, then we have $fc(\{b\}) \sqsubseteq \{c\} \in SI$, which is contradictory and should be omitted.
      iii. If X' is of form $fc(\{c\})$, then similar as the proof form situation **R4** of invariant 57 we have $fc(\{b\}) \sqsubseteq fc(\{c\}) \in SC$. Due to the non-applicability of **AR4a** we have $a : B \in SC$.
3. If it is derived by **R11**, then we have $A_1, \ldots, A_n$ s.t. $B = fc(A_i)$ and $\{a\} \sqsubseteq A_j \in SI$ $(2 \leq j \leq n, j \neq i)$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq \bot \in \mathcal{T}$. Given the form of the later, it should also be in $\mathcal{T}$. Similar as the proof for situation **R2**, due to the non-applicability of **AR11** we have $a : B \in SC$.
4. If it is derived by **R12**, then there is individual $b \in \mathcal{IN_A}$ and role $r \in \mathcal{RN_T}$ s.t. $\{a\} \sqsubseteq \exists r.\{b\} \in SI$ and $fc(B) \sqsubseteq \exists fc(r).\{b\} \in SI$. By induction of invariant 66 we have $(a, b) : r \in SC$. By induction of invariant 60 we have $fc(B) \sqsubseteq \exists fc(r).\{b\} \in SC$. Due to the non-applicability of **CR12** we have $a : B \in SC$.

Thus the induction of invariant 61 is proved.

Invariant 62: $\{a\} \sqsubseteq \{b\}$ can be derived by **R1, 6, 10, 11, 12**. Note that **R2** and **R4** can not derive $\{a\} \sqsubseteq \{b\}$ because $\{b\}$ should not appear as the RHS in $\mathcal{T}$.

1. If it is derived by **R1**, then there is a concept X s.t. $\{a\} \sqsubseteq X \in SI$ and $X \sqsubseteq \{b\} \in SI$. We make a case distinction on the form of X:
   (a) If $X, fc(X) \in \mathcal{CN}_\mathcal{T}$, given the possible forms of axioms in SI, we have $X \sqsubseteq \bot \in SI$. By induction of invariant 61 we have $a : X \in SC$. By induction of invariant 57 we have $X \sqsubseteq \bot \in ST \subseteq SC$. Due to the non-applicability of **AR1a** we have $a : \bot \in SC$. Due to the non-applicability of **AR1e** we have $a \doteq b \in SC$.
   (b) If X is a nominal $\{c\}$. Thus by induction of invariant 62 we have $a \doteq c \in SC$ and $c \doteq b \in SC$. By non-applicability of **AR1d** we have $a \doteq b \in SC$.
   (c) If X is of form $fc(\{c\})$, then we have $fc(\{c\}) \sqsubseteq \{b\} \in SI$, which is contradictory to the possible forms of axioms in SI thus we omit this situation.
2. If it is derived by **R6**, then there is $\{b\} \sqsubseteq \{a\} \in SI$. As we explained earlier in the proof, if $\{b\} \sqsubseteq \{a\}$ is an initial element of SI without any inference, then we have $b \doteq a \in SC$, due to the non-applicability of **IR3**. Alternatively, if $\{b\} \sqsubseteq \{a\} \in SI$ is derived, then by induction of invariant 62 we have $b \doteq a \in SC$. In either case, by non-applicability of **AR10** we have $a \doteq b \in SC$.
3. If it is derived by **R10**, then there is $fc(\{b\}) \sqsubseteq fc(\{a\}) \in SI$. By induction of invariant 63 we have $b \doteq a \in SC$. Due to the non-applicability of **AR10** we have $a \doteq b \in SC$.
4. If it is derived by **R11**, then we have $A_1, \ldots, A_n$ s.t. $\{b\} = fc(A_i)$ and $\{a\} \sqsubseteq A_j \in SI(1 \leq j \leq n, j \neq i)$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq \bot \in \mathcal{T}$. Similar as the proof for situation **R11** of invariant 61, due to the non-applicability of **AR11** we have $a : \{b\} \in SC$. Due to the non-applicability of **IR6** we have $a \doteq b \in SC$.
5. If it is derived by **R12**, then there is individual $c \in \mathcal{IN}_\mathcal{A}$ and role $r \in \mathcal{RN}_\mathcal{T}$ s.t. $\{a\} \sqsubseteq \exists r.\{c\} \in SI$ and $fc(\{b\}) \sqsubseteq \exists fc(r).\{c\} \in SI$. By induction of invariant 66 we have $(a, c) : r \in SC$. $fc(\{b\}) \sqsubseteq \exists fc(r).\{c\} \in SI$ is contradictory to the possible forms of axioms in SI unless we have $\top \sqsubseteq \exists fc(r).\{c\} \in SI$. By induction of invariant 60 we have $\top \sqsubseteq \exists fc(r).\{c\} \in SC$. Because we have $a : \top \in SC$, due to the non-applicability of **AR3a** we have $a : \exists fc(r).\{c\} \in SC$. Due to the non-applicability of **IR8** we have $(a, c) : fc(r) \in SC$. Due to the non-applicability of **AR12** we have $a \not\doteq a \in SC$. Because we also have $a \doteq a \in SC$, due to the non-applicability of **AR9b** we have $a : \bot \in SC$. Finally, due to the non-applicability of **AR1e** we have $a \doteq b \in SC$.

Thus the induction of invariant 62 is proved.

Invariant 63: Similar as invariant 58, $fc(\{a\}) \sqsubseteq fc(\{b\})$ can be derived by **R1, 2, 4, 10, 11, 12**. We make a case distinction as follows:

1. If it is derived by **R1**, then there is a concept X s.t. $fc(\{a\}) \sqsubseteq X \in SI$ and $X \sqsubseteq fc(\{b\}) \in SI$. We make a case distinction on form of X:
   (a) If $X, fc(X) \in \mathcal{CN}_\mathcal{T}$, given the possible form of axioms in SI, we have $\top \sqsubseteq X \in SI$. By induction of invariant 36 we have $\top \sqsubseteq X \in SC$. Due to the non-applicability of **R10** we have $fc(X) \sqsubseteq \bot \in SC$. By induction of invariant 37 we have $b : fc(X) \in SC$. Together due to the non-applicability of **AR1a** we have $b : \bot \in SNF$. Due to the non-applicability of **AR1e** and **AR10** we eventually have $a \doteq b \in SC$.
   (b) If X is a nominal $\{c\}$, then we have $fc(\{a\}) \sqsubseteq \{c\} \in SI$, which contradicts with the possible forms of axioms in SI thus we omit this situation.
   (c) If X is of form $fc(\{c\})$. Thus by induction of invariant 41 we have $a \doteq c \in SC$ and $c \doteq b \in SC$. By non-applicability of **AR1d** we have $a \doteq c \in SC$.
2. If it is derived by **R2**, then we have $A_1, \ldots, A_n$ s.t. $fc(\{a\}) \sqsubseteq A_i \in SI$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq fc(\{b\}) \in \mathcal{T}$. We make a case distinction on the form of A_i:

(a) If $A_i, fc(A_i) \in \mathcal{CN}_\mathcal{T}$, given the form of axiom we have $\top \sqsubseteq A_i \in SI$. By induction of invariant 57 we have $\top \sqsubseteq A_i \in SC$.

(b) If A_i is a nominal $\{c\}$ then we have $fc(\{a\}) \sqsubseteq \{c\} \in SI$ which is contradictory to the possible forms of axioms in SI thus we omit this situation.

(c) If A_i is of form $fc(\{c\})$, then by induction of invariant 63 we have $a \doteq c \in SC$.

Combining the above possibilities, due to the non-applicability of **CR2** we have $a \doteq b \in SC$.

3. If it is derived by **R4**, then there is $r \in \mathcal{RN}_\mathcal{T}$ and two concepts X and X' s.t. $fa(\{a\}) \sqsubseteq \exists r.X \in SI$, $X \sqsubseteq X' \in SI$ and $\exists r.X' \sqsubseteq fc(\{b\}) \in \mathcal{T}'$. Given the form of $\exists r.X' \sqsubseteq fc(\{b\})$, it should also be in $\mathcal{T}$. Regardless the form of X, given the possible forms of axioms in SI, we always have $\top \sqsubseteq \exists r.X \in SI$. Regardless the form of X' is, we can always prove that $b : \bot \in SC$ as in the proof for situation **R4** of invariant 58. Hence due to the non-applicability of **AR1e** we have $a \doteq b \in SC$.
4. If it is derived by **R10**, then there is $\{b\} \sqsubseteq \{a\} \in SI$. By induction of invariant 62 we have $b \doteq a \in C$. Due to the non-applicability of **AR10** we have $a \doteq b \in SC$.
5. If it is derived by **R11**, then we have $A_1, \ldots, A_n$ s.t. $A_i = \{b\}$ and $fc(\{a\}) \sqsubseteq A_j \in SI$ $(1 \leq j \leq n, j \neq i)$ and $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq \bot \in \mathcal{T}$. We make a case distinction on the form of A_j as follows:

   (a) If $A_j, fc(A_j) \in \mathcal{CN}_\mathcal{T}$, then $fc(\{a\}) \sqsubseteq A_j$ is contradictory to the possible forms of axioms in SI unless we have $T \sqsubseteq A_j$.

   (b) If A_j is a nominal $\{c\}$, then it is contradictory to the possible forms of axioms in SI so we omit this situation.

   (c) If A_j is of form $fc(\{c\})$, then by induction of invariant 63 we have $a \doteq c \in SC$.

   Together, due to the non-applicability of **CR11** we have $a : \{b\} \in SC$. Note that, here we don't introduce new nominal because $\{b\}$ is native. Due to the non-applicability of **IR6** we have $a \doteq b \in SC$.
6. If it is derived by **R12**, then there is individual $c \in \mathcal{IN}_\mathcal{A}$ and role $r \in \mathcal{RN}_\mathcal{T}$ s.t. $fc(\{a\}) \sqsubseteq \exists r.\{c\} \in SI$ and $\{b\} \sqsubseteq \exists fc(r).\{c\} \in SI$. Then follow a similar proof for situation **R12** of invariant 62 we can prove that $a \doteq b \in SC$.

Thus the induction of invariant 63 is proved.

Invariant 64: $r \sqsubseteq s \in SI$ can be derived by **R14, 15, 16, 18**. The situation of **R14, 15** and **18** so we skip them.

When it is derived by **R18**, then we have $\exists r.\top \sqsubseteq X$, $\exists fi(r).\top \sqsubseteq Y$, $\bigsqcap_{1 \leq i \leq n} fc(\{a_i\}) \sqsubseteq fc(A)$ and $\bigsqcap_{1 \leq j \leq m} fc(\{b_j\}) \sqsubseteq fc(B) \in \mathcal{T}'$. Given the form of these axioms, they should also be in $\mathcal{T}$. We also have $X \sqsubseteq A$, $Y \sqsubseteq B$, $fc(A) \sqsubseteq fc(\{a_i\})$ $(1 \leq i \leq n)$, $fc(B) \sqsubseteq fc(\{b_j\})$ $(1 \leq j \leq m)$ and $\{a_i\} \sqsubseteq \exists s.\{b_j\} \in SI$ $(1 \leq i \leq n, 1 \leq j \leq m)$. By induction of invariant 66 we should have $(a_i, b_j) : s \in SC$ $(1 \leq i \leq n, 1 \leq j \leq m)$. Note that all the $\{a_i\}$ and $\{b_j\}$ are native. Hence we should have $\{a_i\} \sqsubseteq \exists s.\{b_j\} \in SC$ $(1 \leq i \leq n, 1 \leq j \leq m)$. Now we make a case distinction on the form of X and A:

1. If $A, fc(A) \in \mathcal{CN}_\mathcal{T}$, by induction of invariant 58 we have $a_i : A \in SC$. Given that $\{a_i\}$ and $fc(\{a_i\})$ are native, due to the non-applicability of **IR1** and **R10** we have $fc(A) \sqsubseteq fc(\{a_i\}) \in SC$ $(1 \leq i \leq n)$:

   (a) If $X, fc(X) \in \mathcal{CN}_\mathcal{T}$, by induction of invariant 57 we have $X \sqsubseteq A \in SC$.

   (b) If X is a nominal $\{c\}$, then we have $\exists r.\top \sqsubseteq \{c\} \in SI$, which is contradictory to the possible forms of axioms in SI so we omit this situation.

   (c) If X is of form $fc(\{c\})$, then we have $fc(\{c\}) \sqsubseteq A \in SI$, which is contradictory to the possible forms of axioms in SI unless we have $\top \sqsubseteq A \in SI$. By induction of invariant 57 we have $\top \sqsubseteq A \in SC$. Given that $fc(\{c\})$ is native, we have $fc(\{c\}) \sqsubseteq \top \in SC$. Together due to non-applicability of **R1** we have $X \sqsubseteq A \in SC$.

2. If A is a nominal $\{c\}$, by induction of invariant 63 we have $c \doteq a_i \in SC$. Given the $\{a_i\}$, $fc(\{a_i\})$, $\{c\}$ and $fc(\{c\})$ are all native. We should have $fc(\{c\}) \sqsubseteq fc(\{a_i\}) \in SC$ $(1 \leq i \leq n)$:
   (a) If $X, fc(X) \in \mathcal{CN}_{\mathcal{T}}$, then we have $X \sqsubseteq \{c\} \in SI$, which is contradictory to the possible forms of axioms in SI unless we have $X \sqsubseteq \bot \in SI$. In this case, by induction of invariant 57 we have $X \sqsubseteq \bot \in SC$. Due to the non-applicability of **R10** we have $\top \sqsubseteq fc(X) \in SC$. Given than $fc(\{c\})$ is native, we have $fc(\{c\}) \sqsubseteq \top \in SC$. Together we have $fc(\{c\}) \sqsubseteq fc(X) \in SC$ and furthermore $X \sqsubseteq \{c\} \in SC$.
   (b) If X is a nominal $\{d\}$, then by induction we have $d \doteq c \in SC$. Given that both $\{d\}$ and $\{c\}$ are native, we have $X \sqsubseteq A = \{d\} \sqsubseteq \{c\} \in SC$.
   (c) If X is of form $fc(\{d\})$, then we have $fc(\{d\}) \sqsubseteq \{c\} \in SI$, which is contradictory to the possible forms of axioms in SI thus we omit this situation.

3. If A is of form $fc(\{c\})$, by induction of invariant 65 we have $c \not\doteq a_i \in SC$ $(1 \leq i \leq n)$. Given that $\{c\} = fc(A)$ is native, and $\{a_i\}$ is native, eventually we have $\{c\} \sqsubseteq fc(\{a_i\}) \in SC$ $(1 \leq i \leq n)$:
   (a) If $X, fc(X) \in \mathcal{CN}_{\mathcal{T}}$, we have $X \sqsubseteq fc(\{c\}) \in SI$. By induction of invariant 58 we have $c : fc(X) \in SC$. Given that $\{c\}$ is native we should have $X \sqsubseteq fc(\{c\}) \in SC$.
   (b) If X is a nominal $\{d\}$, by induction of invariant 65 we have $d \not\doteq c \in SC$. Given that $\{d\}$ is also native, eventually we should have $\{d\} \sqsubseteq fc(\{c\}) \in SC$.
   (c) If X is of form $fc(\{d\})$, by induction of invariant 63 we have $d \doteq c \in SC$. Given that $fc(\{d\})$ and $fc(\{c\})$ are both native, eventually we should have $fc(\{d\}) \sqsubseteq fc(\{c\}) \in SC$.

Hence in all the above situations we have $X \sqsubseteq A \in SC$ and $fc(A) \sqsubseteq fc(\{a_i\}) \in SC$ $(1 \leq i \leq n)$. Similarly, we can also prove that we always have $Y \sqsubseteq B \in SC$ and $fc(B) \sqsubseteq fc(\{b_j\}) \in SC$ $(1 \leq j \leq m)$. Together, due to the non-applicability of **R18** we have $r \sqsubseteq s \in SC$.

Thus the induction of invariant 64 is proved.

Invariant 65: $\{a\} \sqsubseteq fc(\{b\})$ can be derived by **R1, 2, 4, 10, 11, 12, 20**.

Situations **R1, 10, 12** can be proved in a similar way as in proof of invariant 40 so we skip them.

If it is derived by **R2**, **R4** or **R11**, we can prove in a similar manner as situations **R2, 4, 11** of invariant 61, that $a : fc(\{b\}) \in SC$. Due to the non-applicability of **IR7** we have $a \not\doteq b \in SC$.

We also skip the obvious situation where $\{a\} \sqsubseteq \bot \in SI$ is derived before the corresponding rule is applied, in which case we have $a : \bot \in SC$ by induction of invariant 61. Then due to the non-applicability of **AR1e** we can derive $a \not\doteq b \in SC$.

If it is derived by **R20**, then there is concept X s.t. $\{a\} \sqsubseteq \exists r.X, X \sqsubseteq \exists fi(fc(r)).\{b\} \in SI$. We make a case distinction on the form of X:

1. If $X, fc(X) \in \mathcal{CN}_{\mathcal{T}}$, by induction of invariant 68 we have $a : \exists r.X \in SC$. By induction of invariant 60 we have $X \sqsubseteq\sqsubseteq \exists fi(fc(r)).\{b\} \in SC$. Together due to the non-applicability of **CR20** we have $a \not\doteq b \in SC$.

2. If X is a nominal $\{c\}$, by induction of invariant 44 we have $(a,c) : r \in SC$ and $(c,b) : fi(fc(r)) \in SC$. Due to the non-applicability of **AR19** we have $(c,a) : fi(r) \in SC$. Because $fi(fc(r)) = fc(fi(r))$, due to the non-applicability of **AR12** we have $a \neq b \in SC$.
3. If X is of form $fc(\{c\})$, by induction of invariant 66 we have $a : \exists r.fc(\{c\}) \in SC$, which also implies that $fc(\{c\})$ is native. We also have $fc(\{c\}) \sqsubseteq \exists fi(fc(r)).\{b\} \in SI$, which is contradictory to the possible forms of axioms in SI unless $\top \sqsubseteq \exists fi(fc(r)).\{b\} \in SI$. By induction of invariant 60 we have $\top \sqsubseteq \exists fi(fc(r)).\{b\} \in SC$. Because $fc(\{c\})$ is native, we have $fc(\{c\}) \sqsubseteq \top \in SC$. Due to the non-applicability of **R3** we have $fc(\{c\}) \sqsubseteq \exists fi(fc(r)).\{b\} \in SC$. Due to the non-applicability of **CR20** we have $a \neq b \in SC$.

Thus the induction of invariant 65 is proved.

Invariant 66: $\{a\}\exists r.\{b\} \in SI$ can be derived by **R3, 7, 8, 19**. We make a case distinction as follows:

1. If it is derived by **R3**, then there is concept X s.t. $\{a\} \sqsubseteq X, X \sqsubseteq \exists r.\{b\} \in SI$. We make a case distinction on the form of X:
   (a) If $X, fc(X) \in \mathcal{CN}_\mathcal{T}$, by induction of invariant 61 we have $a : X \in SC$. By induction of invariant 60 we have $X \sqsubseteq \exists r.\{b\} \in SC$. Due to the non-applicability of **AR3a** we have $a : \exists r.\{b\} \in SC$. Due to the non-applicability of **IR8** we have $(a,b) : r \in SC$.
   (b) If X is a nominal $\{c\}$, by induction of invariant 62 we have $a \doteq c \in SC$. By induction of invariant 66 we have $(c,b) : r \in SC$. Due to the non-applicability of **AR3c** we have $(a,b) : r \in SC$.
   (c) If X is of form $fc(\{c\})$, then we have $fc(\{c\}) \sqsubseteq \exists r.\{b\} \in SI$, which is contradictory to the possible forms of axioms in SI unless we have $\top \sqsubseteq \exists r.\{b\} \in SI$. In this situation we can easily derive that $a : \exists r.\{b\} \in SC$ and hence $(a,b) : r \in SC$.
2. If it is derived by **R7**, then there is $s \in \mathcal{RN}_\mathcal{T}$ s.t. $\{a\} \sqsubseteq \exists s.\{b\}, s \sqsubseteq r \in SI$. By induction of invariant 66 we have $(a,b) : s \in SC$. By induction of invariant 64 we have $s \sqsubseteq r \in SC$. Due to the non-applicability of **AR7b** we have $(a,b) : r \in SC$.
3. If it is derived by **R8**, then there are a concept X and roles $r_1, r_2 \in \mathcal{RN}_\mathcal{T}$ s.t. $\{a\} \sqsubseteq \exists r_1.X, X \sqsubseteq \exists r_2.\{b\}, r_1 \circ r_2 \sqsubseteq r \in SI$. We make a case distinction on the form of X:
   (a) If $X, fc(X) \in \mathcal{CN}_\mathcal{T}$, then by induction of invariant 68 we have $a : \exists r_1.X \in SC$. By induction of invariant 60 we have $X \sqsubseteq \exists r_2.\{b\} \in SC$. Due to the non-applicability of **CR8b** we have $(a,b) : r \in SC$.
   (b) If X is a nominal $\{c\}$. By induction of invariant 66 we have $(a,c) : r_1, (c,b) : r_2 \in SC$. Due to the non-applicability of **AR8c** we have $(a,b) : r \in SC$.
   (c) If X is of form $fc(\{c\})$, then by induction of invariant 68 we have $a : \exists r_1.fc(\{c\}) \in SC$, which also indicates that $fc(\{c\})$ is native. Hence we have $fc(\{c\}) \sqsubseteq \top \in SC$. We also have $fc(\{c\}) \sqsubseteq \exists r_2.\{b\} \in SI$, which is contradictory to the possible forms of axioms in SI unless we have $\top \sqsubseteq \exists r_2.\{b\} \in SI$. In this case by induction of invariant 60 we have $\top \sqsubseteq \exists r.\{b\} \in SC$. Due to the non-applicability of **R3** we have $fc(\{c\}) \sqsubseteq \exists r.\{b\} \in SC$. Due to the non-applicability of **AR8a** we have $a : \exists r.\{b\} \in SC$. Due to the non-applicability of **IR8** we have $(a,b) : r \in SC$.
4. If it is derived by **R19**, then we have $\{b\} \sqsubseteq \exists fi(r).\{a\} \in SI$. By induction of invariant 66 we have $(b,a) : fi(r) \in SC$. Due to the non-applicability of **AR19** we have $(a,b) : r \in SC$.

Thus the induction of invariant 66 is proved.

Invariant 67: $r_1 \circ r_2 \sqsubseteq s \in SI$ can only be derived by **R17**. It is easy to show that by induction of invariant 67 itself and the non-applicability of **R17** we have $r_1 \circ r_2 \sqsubseteq s \in SC$. Thus the induction of invariant 67 is proved.

Invariant 68: $\{a\} \sqsubseteq \exists r.B \in SI$ can be derived by **R3, 7, 8**. Note that $\{a\} \sqsubseteq \bot \notin SI$. We make a case distinction as follows:

1. If it is derived by **R3**, then there is a concept X s.t. $\{a\} \sqsubseteq X, X \sqsubseteq \exists r.B \in SI$. We make a case distinction on form of X:
   (a) If $X, fc(X) \in \mathcal{CN}_\mathcal{T}$, then by induction of invariant 61 we have $a : X \in SC$. By induction of invariant 42 we have $X \sqsubseteq \exists r.B \in SC$. Due to the non-applicability of rule **AR3a** we have $a : \exists r.B \in SC$.
   (b) If X is a nominal $\{b\}$, then by induction of invariant 44 we have $a \doteq b \in SC$. By induction of invariant 43 we have $b : \exists r.B \in SC$. Due to the non-applicability of **AR3b** we have $a : \exists r.B \in SC$.
   (c) If X is of form $fc(\{b\})$, then we have $fc(\{b\}) \sqsubseteq \exists r.B \in SI$, which is contradictory to possible forms of axioms in SI unless we have $\top \sqsubseteq \exists r.B \in SI$. In this case by induction of invariant 59 we have $\top \sqsubseteq \exists r.B \in SC$. Because we have $a : \top \in SC$. Due to the non-applicability of **AR3a** we have $a : \exists r.B \in SC$.
2. If it is derived by **R7**, there is $s \in \mathcal{RN}_\mathcal{T}$ s.t. $\{a\} \sqsubseteq \exists s.B \in SI$ and $s \sqsubseteq r \in SI$. By induction of invariant 68 we have $a : \exists s.B \in SC$. By induction of invariant 64 we have $s \sqsubseteq r \in SC$. Due to non-applicability of **AR7a** we have $a : \exists r.B \in SC$.
3. If it is derived by **R8**, there are concept X and roles $r_1, r_2 \in \mathcal{RN}_\mathcal{T}$ s.t. $\{a\} \sqsubseteq \exists r_1.X, X \sqsubseteq \exists r_2.B, r_1 \circ r_2 \sqsubseteq r \in SI$. By induction of invariant 67 we have $r_1 \circ r_2 \sqsubseteq r \in SC$. We make a case distinction on form of X:
   (a) If $X, fc(X) \in \mathcal{CN}_\mathcal{T}$, by induction of invariant 68 we have $a : \exists r_1.X \in SC$. By induction of invariant 59 we have $X \sqsubseteq \exists r_2.B \in SC$. Due to the non-applicability of **AR8a** we have $a : \exists r.B \in SC$.
   (b) If X is a nominal $\{b\}$, by induction of invariant 66 we have $(a, b) : r_1 \in SC$. By induction of invariant 68 we have $b : \exists r_2.B \in SC$. Due to the non-applicability of **AR8b** we have $a : \exists r.B \in SC$.
   (c) If X is of form $fc(\{b\})$, then following a similar proof for situation **R8** of invariant 66 we have $a : \exists r.B \in SC$.

Thus the induction of invariant 68 is proved.

With inductions of all invariants proved for each of the corresponding rules, we proved Lemma 16.

This further proves the → directions of the Theorem and hence the Theorem 15. □

Theorem 17 *Let $\mathcal{O}$ be an $\mathcal{ELHI}$ TBox with only safe roles and $TA_{fa}(\mathcal{O})$ its TBox transformation with name assignment $fa(.)$, then for any $A, B \in \mathcal{CN}_\mathcal{O}$, $\mathcal{O} \models A \sqsubseteq B$* iff *$A \sqsubseteq B \in ST(TA_{fa}(\mathcal{O}))$ or $A \sqsubseteq \bot \in ST(TA_{fa}(\mathcal{O}))$.*

Proof The ← direction immediately follows from the soundness of our approach (Theorem 5) therefore we will only need to prove the completeness.

The → direction can be proved with model construction and contrapositive. For the sake of conciseness, in this proof we use ST to abbreviate $ST(TA_{fa}(\mathcal{O}))$. Particularly, assuming there are $X, Y \in \mathcal{CN}_{\mathcal{O}}$ s.t. $\mathcal{O} \models X \sqsubseteq Y$, $X \sqsubseteq Y \notin ST$ and $X \sqsubseteq \bot \notin ST$, we want to show that a model $\mathcal{I}$ of $\mathcal{O}$ can be constructed such that $\mathcal{I} \not\models X \sqsubseteq Y$.

In order to construct such a model, we define a relation $\approx$ on ST s.t. $A \approx B$ *iff* $A \equiv B \in ST$ (or $A \sqsubseteq B \in ST$ when $B = \bot$). Obviously $\approx$ is an equivalence relation. We use $\approx$ to partition the $\mathcal{CN}_{ST}$ and use $[A]$ to denote the set $\{B | A \approx B\}$. Obviously, the following holds:

1. If $A \sqsubseteq B \in ST$, then for any $A' \in [A]$, we have $A' \sqsubseteq B \in ST$. And for any $B' \in [B]$, we have $A \sqsubseteq B' \in ST$.
2. If $A \sqsubseteq \exists r.B \in ST$, then for any $A' \in [A]$, we have $A' \sqsubseteq \exists r.B \in ST$.

These two properties will ensure the well-definition of the model. Simply, we also use $[\bot]$ to denote all the unsatisfiable concepts in ST.

We also show that for roles R, S that occur in $\mathcal{O}$, we have $R \preccurlyeq^* S$ if we have $fa(R) \sqsubseteq fa(S) \in ST$. This can be shown by induction:

1. For each original RIA $fa(R) \sqsubseteq fa(S)$ in the transformed TBox, we know that there is $R \sqsubseteq S \in \mathcal{O}$. Hence $R \preccurlyeq^* S$.
2. For each tautology RIA $fa(R) \sqsubseteq fa(R)$ in the initialisation of ST, we have $R = R$ hence $R \preccurlyeq^* R$.
3. If $fa(R) \sqsubseteq fa(S)$ is derived by rule **R14**, then there is some $fa(T)$ s.t. $fa(R) \sqsubseteq fa(T) \in ST$ and $fa(T) \sqsubseteq fa(S) \in ST$. Inductively, we know that $R \preccurlyeq^* T$ and $T \preccurlyeq^* S$. Hence $R \preccurlyeq^* S$.
4. If $fa(R) \sqsubseteq fa(S)$ is derived by rule **R15**, then there is $fi(fa(R)) \sqsubseteq fi(fa(S)) \in ST$. Note that $fi(fa(R)) = fa(inv(R))$ and $fi(fa(S)) = fa(inv(S))$, we have $fa(inv(R)) \sqsubseteq fa(inv(S)) \in ST$. Inductively, there is $inv(R) \preccurlyeq^* inv(S)$. According to the definition of safe role we have $R \preccurlyeq^* S$.

Now assume there are $X, Y \in \mathcal{CN}_{\mathcal{O}}$ s.t. $\mathcal{O} \models X \sqsubseteq Y$, $X \sqsubseteq Y \notin ST$ and $X \sqsubseteq \bot \notin ST$. We know that X must be satisfiable in ST. This implies that a model $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ constructed as follows should exist with a non-empty $\Delta^{\mathcal{I}}$:

1. $\Delta^{\mathcal{I}} = \{[A] | A \notin [\bot]\}$;
2. $A^{\mathcal{I}} = \{[B] | B \sqsubseteq A \in ST\} \setminus [\bot]$;
3. $r^{\mathcal{I}} = direct(r) \cup inverse(r)$, where:
   (a) $direct(r) = \{\langle [A], [B] \rangle | A \sqsubseteq \exists s.B \in ST, s \sqsubseteq r \in ST\}$;
   (b) $inverse(r) = \{\langle [A], [B] \rangle | \langle [B], [A] \rangle \in direct(fi(r))\}$;

Apparently, since $X \sqsubseteq X \in ST$, we have $[X] \in X^{\mathcal{I}}$. Also, $X \sqsubseteq Y \notin ST$, therefore $[X] \notin Y^{\mathcal{I}}$. Hence, $\mathcal{I} \not\models X \sqsubseteq Y$.

Now we show that $\mathcal{I}$ is indeed a model of $\mathcal{O}$. This can be done by analysing each axiom $\alpha \in \mathcal{O}$ in a case by case manner:

1. If α is of form $A \sqsubseteq B$, then obviously such an axiom will be preserved after transformation and be included in ST. For any $[C] \in A^{\mathcal{I}}$, we know that $C \sqsubseteq A \in ST$ as well. Due to the non-applicability of rule **R1**, we have $X \sqsubseteq A \in ST$, which implies that $[X] \in B^{\mathcal{I}}$. Hence, $\mathcal{I} \models \alpha$.
2. If α is of form $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq B$, then this axiom will be transformed into the following two axioms during TBox transformation:

$$fa(A_1 \sqcap \cdots \sqcap A_n) \sqsubseteq B \tag{112}$$

$$fa(A_1 \sqcap \cdots \sqcap A_n) \equiv A_1 \sqcap \cdots \sqcap A_n \tag{113}$$

Obviously, both of these axioms will be included in ST as well. For any $[C] \in (A_1 \sqcap \cdots \sqcap A_n)^{\mathcal{I}}$, by the semantics of conjunction we know that $[C] \in A_i^{\mathcal{I}}$ for each $i = 1, \ldots, n$. Hence, we have $C \sqsubseteq A_i \in ST$. Due to the non-applicability of rule **R2** and axiom (113), we have $C \sqsubseteq fa(A_1 \sqcap \cdots \sqcap A_n) \in ST$. Further with axiom (112) we have $C \sqsubseteq B \in ST$. This implies that $[C] \in B^{\mathcal{I}}$ and $\mathcal{I} \models \alpha$.

3. If α is of form $A \sqsubseteq \exists R.B$, then the following axioms will be created in the TBox transformation:

$$A \sqsubseteq fa(\exists R.B) \tag{114}$$

$$fa(\exists R.B) \equiv \exists fa(R).B \tag{115}$$

Similar as before, these two axioms will also be included in ST. For any $[C] \in A^{\mathcal{I}}$, we have $C \sqsubseteq A \in ST$. Due to the non-applicability of rule **R3** and the above two axioms in ST, we will have $C \sqsubseteq \exists fa(R).B \in ST$. Since $[C] \in \Delta^{\mathcal{I}}$ we know that $C \notin [\bot]$. Consequently we know that $B \notin [\bot]$. Hence, $[B] \in B^{\mathcal{I}}$. Also, $C \sqsubseteq \exists fa(R).B \in ST$ suggests that $\langle [C], [B] \rangle \in direct(fa(R)) \subseteq fa(R)^{\mathcal{I}}$. Now we make a case distinction on R:

   (a) When $R \in \mathcal{RN}_{\mathcal{O}}$, we know that $fa(R) = R$. Hence, $[B] \in B^{\mathcal{I}}$ and $\langle [C], [B] \rangle \in R^{\mathcal{I}}$ implies that $[C] \in (\exists R.B)^{\mathcal{I}}$.

   (b) When $R \notin \mathcal{RN}_{\mathcal{O}}$, we know that R is the inverse role of some named role in $\mathcal{RN}_{\mathcal{O}}$. Without lose of generality, assuming $R = r^{-}$, we know that $fa(R) = fi(r)$ and $r = fi(fa(R))$ in ST. Given that $\langle [C], [B] \rangle \in direct(fa(R))$, we know that $\langle [B], [C] \rangle \in inverse(r) \subseteq r^{\mathcal{I}}$. Hence, $\langle [C], [B] \rangle \in R^{\mathcal{I}}$. Together with $[B] \in B^{\mathcal{I}}$, we know that $[C] \in (\exists R.B)^{\mathcal{I}}$.

This shows that $\mathcal{I} \models \alpha$ in all possible cases.

4. If α is of form $\exists R.A \sqsubseteq B$, then the following axioms will be created in the TBox transformation:

$$fa(\exists R.A) \sqsubseteq B \tag{116}$$

$$fa(\exists R.A) \equiv \exists fa(R).A \tag{117}$$

These two axioms will also be included in ST. For any $[C] \in fa(\exists R.A)^{\mathcal{I}}$, by definition of DL semantics, we know that there is some $[D] \in A^{\mathcal{I}}$ s.t. $\langle [C], [D] \rangle \in R^{\mathcal{I}}$. From the former we have $D \sqsubseteq A \in ST$. For the latter we make a case distinction on R:

   (a) When $R \in \mathcal{RN}_{\mathcal{O}}$, we have $fa(R) = R$. We can make a further case distinction on $\langle [C], [D] \rangle R^{\mathcal{I}}$:

      i. If $\langle [C], [D] \rangle \in direct(R)$, then we have $C \sqsubseteq \exists s.D \in ST$ and $s \sqsubseteq R \in ST$. Combining with $D \sqsubseteq A \in ST$, axiom (117), due to the non-applicability of rule **R4**, we have $C \sqsubseteq fa(\exists R.A) \in ST$ and furthermore $C \sqsubseteq B \in ST$. The latter implies that $[C] \in B^{\mathcal{I}}$.

      ii. If $\langle [C], [D] \rangle \in inverse(R)$, then we have $\langle [D], [C] \rangle \in direct(fi(R))$. This suggests that there is some role s s.t. $D \sqsubseteq \exists s.C \in ST$ and $s \sqsubseteq fi(R) \in ST$. Note that $s \sqsubseteq fi(R) \in ST$ indicates that $s \preccurlyeq inv(R)$. This suggests that s is not a safe role and is against the condition in the theorem. Hence, such a situation should not occur.

(b) When $R \notin \mathcal{RN}_{\mathcal{O}}$, we know that R is the inverse of some named role in $\mathcal{O}$. Without lose of generality, we assume $R = r^-$. Then we have $fa(R) = fi(r)$ and $r = fi(fa(R))$ in ST. This also suggests that $\langle [D], [C] \rangle \in r^{\mathcal{I}}$. We can make a further case distinction:

i. If $\langle [D], [C] \rangle \in direct(r)$, then there should be some s in ST s.t. $D \sqsubseteq \exists s.C \in ST$ and $s \sqsubseteq r \in ST$. The later suggests that $s \preccurlyeq r$. In other words, $s \preccurlyeq inv(r^-)$ and $\exists r^-.A \sqsubseteq B \in \mathcal{O}$. This implies that s is not a safe role and it is against the condition in the theorem. Hence, such a situation should not occur.

ii. If $\langle [D], [C] \rangle \in inverse(r)$, then by definition of $\mathcal{I}$ we have $\langle [C], [D] \rangle \in direct(fa(R))$. This suggests that there is some role s s.t. $C \sqsubseteq \exists s.D \in ST$ and $s \sqsubseteq fa(R) \in ST$. Similar as before we have $C \sqsubseteq B \in ST$ and $[C] \in B^{\mathcal{I}}$.

To sum up, in all possible situations we have $[C] \in B^{\mathcal{I}}$. This indicates that $\mathcal{I} \models \alpha$.

5. If α is of form $R \sqsubseteq S$, then we will have $fa(R) \sqsubseteq fa(S)$ in the transformed TBox and also ST. For any $\langle [C], [D] \rangle \in R^{\mathcal{I}}$, we make the following case distinction:

(a) If $R \in \mathcal{RN}_{\mathcal{O}}$ and $S \in \mathcal{RN}_{\mathcal{O}}$, then $fa(R) = R$ and $fa(S) = S$. We make a further case distinction as follows:

i. If $\langle [C], [D] \rangle \in direct(fa(R))$, then there is some s s.t. $C \sqsubseteq \exists s.D \in ST$ and $s \sqsubseteq fa(R) \in ST$. Together with $fa(R) \sqsubseteq fa(S) \in ST$ and the non-applicability of **R14**, we have $s \sqsubseteq fa(S) \in ST$. This suggests that $\langle [C], [D] \rangle \in direct(fa(S)) \subseteq fa(S)^{\mathcal{I}} = S^{\mathcal{I}}$.

ii. If $\langle [C], [D] \rangle \in inverse(fa(R))$, then we know that $\langle [D], [C] \rangle \in direct(fi(R))$. This implies that there is some role s s.t. $D \sqsubseteq \exists s.C \in ST$ and $s \sqsubseteq fi(R) \in ST$. Given that $R \sqsubseteq S \in ST$, due to the non-applicability of rule **R15**, we have $fi(R) \sqsubseteq fi(S) \in ST$. Hence, we have $s \sqsubseteq fi(S) \in ST$. Together this suggests that $\langle [D], [C] \rangle \in direct(fi(S))$. Consequently we have $\langle [C], [D] \rangle \in inverse(S) \subseteq S^{\mathcal{I}}$.

(b) If $R \notin \mathcal{RN}_{\mathcal{O}}$ and $S \notin \mathcal{RN}_{\mathcal{O}}$, then R and S are both inverse of some atomic roles in $\mathcal{O}$. Without lose of generality, we assume $R = r^-$ and $S = s^-$. We have $fa(R) = fi(r)$ and $fa(S) = fi(s)$. We also know that $\langle [D], [C] \rangle \in r^{\mathcal{I}}$. Now we make a further case distinction as follows:

i. If $\langle [D], [C] \rangle \in direct(r)$, then there is some t s.t. $D \sqsubseteq \exists t.C \in ST$ and $t \sqsubseteq r \in ST$. From $fa(R) \sqsubseteq fa(S) \in ST$ and the non-applicability of rule **R15**, we have $r \sqsubseteq s \in ST$ and furthermore $t \sqsubseteq s \in ST$. This suggests that $\langle [D], [C] \rangle \in direct(s) \subseteq s^{\mathcal{I}}$.

ii. If $\langle [D], [C] \rangle \in inverse(r)$, then we have $\langle [C], [D] \rangle \in direct(fa(R))$. Similar as before, this leads to $\langle [C], [D] \rangle \in direct(fa(S))$ and $\langle [D], [C] \rangle \in inverse(s) \subseteq s^{\mathcal{I}}$.

In both cases, we have $\langle [C], [D] \rangle \in (s^-)^{\mathcal{I}} = S^{\mathcal{I}}$.

(c) If $R \in \mathcal{RN}_{\mathcal{O}}$ and $S \notin \mathcal{RN}_{\mathcal{O}}$, then $fa(R) = R$ and S is an inverse role of some atomic role in $\mathcal{O}$. Without lose of generality, we assume $S = s^-$. Then we have $fa(S) = fi(s)$. Now we make a case distinction as follows:

i. If $\langle [C], [D] \rangle \in direct(R)$, then there is some t s.t. $C \sqsubseteq \exists t.D \in ST$ and $t \sqsubseteq R \in ST$. This leads to $t \sqsubseteq fa(S) \in ST$ and hence $\langle [C], [D] \rangle \in direct(fa(S))$. According to the construction of $\mathcal{I}$, we have $\langle [D], [C] \rangle \in inverse(s) \subseteq s^{\mathcal{I}}$.

ii. If $\langle [C], [D] \rangle \in inverse(R)$, then we have $\langle [D], [C] \rangle \in direct(fi(R))$. This happens when there is some t s.t. $D \sqsubseteq \exists t.C \in ST$ and $t \sqsubseteq fi(R) \in ST$. Since

$fa(R) \sqsubseteq fa(S) \in ST$, we have $fi(R) \sqsubseteq s \in ST$. Hence we have $t \sqsubseteq s \in ST$. This indicates that $\langle [D], [C] \rangle direct(s) \subseteq s^{\mathcal{I}}$.

In both cases, $\langle [C], [D] \rangle \in (s^{-})^{\mathcal{I}} = S^{\mathcal{I}}$.

(d) If $R \notin \mathcal{RN}_{\mathcal{O}}$ and $S \in \mathcal{RN}_{\mathcal{O}}$, then $fa(S) = S$ and R is an inverse role of some atomic role in $\mathcal{O}$. Without lose of generality, we assume $R = r^{-}$. Then we have $fa(R) = fi(r)$ and $\langle [D], [C] \rangle \in r^{\mathcal{I}}$. Now we make a case distinction as follows:

i. If $\langle [D], [C] \rangle \in direct(r)$, then there is some t s.t. $D \sqsubseteq \exists t.C \in ST$ and $t \sqsubseteq r \in ST$. Since $fa(R) \sqsubseteq fa(S) \in ST$, we have $r \sqsubseteq fi(S) \in ST$. Hence $t \sqsubseteq fi(S) \in ST$ and $\langle [D], [C] \rangle \in direct(fi(S))$. This leads to $\langle [C], [D] \rangle \in inverse(S) \subseteq S^{\mathcal{I}}$.

ii. If $\langle [D], [C] \rangle \in inverse(r)$, then we have $\langle [C], [D] \rangle \in direct(fa(R))$. Similar as before, this eventually leads to $\langle [C], [D] \rangle \in direct(fa(S)) = direct(S) \subseteq S^{\mathcal{I}}$.

The above case by case analysis shows that $\mathcal{I} \models \alpha$ when α is an RIA.

Together, we can conclude that $\mathcal{I}$ satisfies each axiom in $\mathcal{O}$. This means that $\mathcal{I}$ is a model of $\mathcal{O}$. Given that $\mathcal{I} \not\models X \sqsubseteq Y$, we know that $\mathcal{O} \not\models X \sqsubseteq Y$. This is contradictory to the assumption that $\mathcal{O} \models X \sqsubseteq Y$, which means that it is not possible to find $X, Y \in \mathcal{CN}_{\mathcal{O}}$ s.t. $\mathcal{O} \models X \sqsubseteq Y$ but $X \sqsubseteq Y \notin ST$. This proves the → direction and the theorem. □