6

Reasoning in the $\mathcal{EL}$ Family of Description Logics

We saw in Chapter 5 that satisfiability and subsumption with respect to general $\mathcal{ALC}$ TBoxes is ExpTime-complete. Interestingly and quite surprisingly, subsumption with respect to general TBoxes is already ExpTime-complete in the small fragment $\mathcal{FL}_0$ of $\mathcal{ALC}$ that has conjunction, *value restriction*, and the top concept as its only concept constructors. In contrast to this negative complexity result for $\mathcal{FL}_0$, subsumption in the description logic $\mathcal{EL}$, which has conjunction, *existential restriction*, and the top concept as its only concept constructors, remains polynomial even in the presence of general TBoxes. Note that, due to the absence of constructors that could cause unsatisfiability, satisfiability is not an interesting inference problem in $\mathcal{FL}_0$ and $\mathcal{EL}$. Also, due to the absence of the complement constructor, subsumption cannot be reduced to unsatisfiability in $\mathcal{FL}_0$ and $\mathcal{EL}$.

The polynomial-time subsumption algorithm for $\mathcal{EL}$ that will be described below differs significantly from the algorithmic techniques for reasoning in DLs introduced in the two previous chapters. For subsumption, the tableau algorithms introduced in Chapter 4 are refutation procedures. In fact, to show that a subsumption $C \sqsubseteq D$ holds, these algorithms refute that a counterexample to the subsumption, i.e., an element of $C \sqcap \neg D$, exists by checking satisfiability of the concept $C \sqcap \neg D$. If $C \sqcap \neg D$ is unsatisfiable, then the subsumption holds, and otherwise it does not hold. The tableau algorithm that tests satisfiability of this concept is nondeterministic due to the presence of disjunction in $\mathcal{ALC}$, and thus an implementation needs to apply backtracking. In contrast, the subsumption algorithm for $\mathcal{EL}$ tries to prove directly that the subsumption holds by iteratively generating GCIs that follow from the TBox. This generation of consequences is deterministic, i.e., any GCI that is generated indeed follows from the TBox, and thus none of the generated consequences needs to be retracted. In the case of $\mathcal{EL}$, there are only polynomially many GCIs that need to be considered as consequences, and the rules generating consequences can be executed in polynomial time. Thus, we obtain a deterministic polynomial time algorithm for subsumption.

This algorithmic approach, which is sometimes called *consequence-based reasoning* in the literature, also turns out to be advantageous for DLs for which subsumption cannot be computed in polynomial time. In fact, for quite a number of interesting DLs (in particular, ones without disjunction and full negation) it is the approach used by highly efficient implemented reasoners. As an example of such a DL, we will consider $\mathcal{ELI}$, the extension of $\mathcal{EL}$ by inverse roles, for which subsumption is known to be ExpTime-complete. In contrast to the tableau algorithm for $\mathcal{ALC}$, the consequence-based algorithm introduced in this chapter is still deterministic, but in the worst case it may, of course, need an exponential amount of time. At first sight, this sounds similar to the type elimination algorithm for satisfiability in $\mathcal{ALC}$ with respect to general TBoxes described in Chapter 5. There is, however, an important difference. The type elimination algorithm starts with the exponentially large set of all types, and then iteratively eliminates types. Consequently, a direct implementation of this approach is also exponential in the best case. In contrast, the subsumption algorithm for $\mathcal{ELI}$ to be introduced below starts with a polynomial number of GCIs that obviously follow from the TBox, and then iteratively adds implied ones. This process may in the worst case generate exponentially many GCIs following from the TBox, but this need not always be the case. In fact, for many practical ontologies, the number of actually implied GCIs is much smaller than the number of possibly implied GCIs.

6.1 Subsumption in $\mathcal{EL}$

The polynomial-time subsumption algorithm for $\mathcal{EL}$ introduced in this section actually classifies a given general TBox $\mathcal{T}$, i.e., it simultaneously computes all subsumption relationships between the concept names occurring in $\mathcal{T}$. Restricting the computation to subsumptions between concept names occurring in $\mathcal{T}$ is without loss of generality since, given compound concepts C, D, we can first add definitions $A \equiv C, B \equiv D$ to the TBox, where A, B are new concept names, and then decide the subsumption $A \sqsubseteq B$ with respect to the extended TBox rather than

$C \sqsubseteq D$ with respect to the original one. The following lemma shows that it is actually sufficient to add only "one half" of each definition.

Lemma 6.1. *Let $\mathcal{T}$ be a general $\mathcal{EL}$ TBox, C, D $\mathcal{EL}$ concepts and A, B concept names not occurring in $\mathcal{T}$ or C, D. Then*

$$\mathcal{T} \models C \sqsubseteq D \quad \textit{if and only if} \quad \mathcal{T} \cup \{A \sqsubseteq C, D \sqsubseteq B\} \models A \sqsubseteq B.$$

Proof. First, assume that $\mathcal{T} \cup \{A \sqsubseteq C, D \sqsubseteq B\} \models A \sqsubseteq B$, and let $\mathcal{I}$ be a model of $\mathcal{T}$. Consider the interpretation $\mathcal{I}'$ that coincides with $\mathcal{I}$ on all role names and all concept names other than A, B, and satisfies $A^{\mathcal{I}'} = C^{\mathcal{I}}$ and $B^{\mathcal{I}'} = D^{\mathcal{I}}$. Since $\mathcal{T}, C, D$ do not contain A, B, the interpretation $\mathcal{I}'$ is a model of $\mathcal{T}$, and it satisfies $C^{\mathcal{I}'} = C^{\mathcal{I}}$ and $D^{\mathcal{I}'} = D^{\mathcal{I}}$. In addition, by the definition of the extensions of A, B in $\mathcal{I}'$, this interpretation is also a model of $\{A \sqsubseteq C, D \sqsubseteq B\}$. Consequently, it satisfies the GCI $A \sqsubseteq B$. Thus, we have $C^{\mathcal{I}} = A^{\mathcal{I}'} \subseteq B^{\mathcal{I}'} = D^{\mathcal{I}}$, which shows that $\mathcal{I}$ satisfies the GCI $C \sqsubseteq D$.

Conversely, assume that $\mathcal{T} \models C \sqsubseteq D$, and let $\mathcal{I}$ be a model of $\mathcal{T} \cup \{A \sqsubseteq C, D \sqsubseteq B\}$. Then, $\mathcal{I}$ is also a model of $\mathcal{T}$, and thus satisfies the GCI $C \sqsubseteq D$. This yields $A^{\mathcal{I}} \subseteq C^{\mathcal{I}} \subseteq D^{\mathcal{I}} \subseteq B^{\mathcal{I}}$, which shows that $\mathcal{I}$ satisfies the GCI $A \sqsubseteq B$. □

6.1.1 Normalisation

To simplify the description of the algorithm, we first transform the given TBox into an appropriate normal form. We say that a general $\mathcal{EL}$ TBox $\mathcal{T}$ is *in normal form* (or *normalised*) if it only contains GCIs of the following form:

$$A \sqsubseteq B, \quad A_1 \sqcap A_2 \sqsubseteq B, \quad A \sqsubseteq \exists r.B, \quad \text{or} \quad \exists r.A \sqsubseteq B,$$

where A, A_1, A_2, B are concept names or the top concept $\top$, and r is a role name. One can transform a given TBox into a normalised one by applying the *normalisation rules* of Figure 6.1. Before showing this for general $\mathcal{EL}$ TBoxes, we illustrate by an example how a given (non-normalised) GCI can be transformed into a set of normalised GCIs using the rules of Figure 6.1:

$$\begin{array}{rcl}
\exists r.A \sqcap \exists r.\exists s.A \sqsubseteq A \sqcap B & \rightsquigarrow_{\mathsf{NF0}} & \underline{\exists r.A \sqcap \exists r.\exists s.A \sqsubseteq B_0}, \quad \underline{B_0 \sqsubseteq A \sqcap B}, \\
\exists r.A \sqcap \exists r.\exists s.A \sqsubseteq B_0 & \rightsquigarrow_{\mathsf{NF1}_\ell} & \exists r.A \sqsubseteq B_1, \quad \underline{B_1 \sqcap \exists r.\exists s.A \sqsubseteq B_0}, \\
B_1 \sqcap \exists r.\exists s.A \sqsubseteq B_0 & \rightsquigarrow_{\mathsf{NF1}_r} & \underline{\exists r.\exists s.A \sqsubseteq B_2}, \quad B_1 \sqcap B_2 \sqsubseteq B_0, \\
\exists r.\exists s.A \sqsubseteq B_2 & \rightsquigarrow_{\mathsf{NF2}} & \exists s.A \sqsubseteq B_3, \quad \exists r.B_3 \sqsubseteq B_2, \\
B_0 \sqsubseteq A \sqcap B & \rightsquigarrow_{\mathsf{NF4}} & B_0 \sqsubseteq A, \quad B_0 \sqsubseteq B.
\end{array}$$

NF0	$\widehat{D} \sqsubseteq \widehat{E}$	$\longrightarrow$	$\widehat{D} \sqsubseteq A,\quad A \sqsubseteq \widehat{E}$
NF1$_r$	$C \sqcap \widehat{D} \sqsubseteq B$	$\longrightarrow$	$\widehat{D} \sqsubseteq A,\quad C \sqcap A \sqsubseteq B$
NF1$_\ell$	$\widehat{D} \sqcap C \sqsubseteq B$	$\longrightarrow$	$\widehat{D} \sqsubseteq A,\quad A \sqcap C \sqsubseteq B$
NF2	$\exists r.\widehat{D} \sqsubseteq B$	$\longrightarrow$	$\widehat{D} \sqsubseteq A,\quad \exists r.A \sqsubseteq B$
NF3	$B \sqsubseteq \exists r.\widehat{D}$	$\longrightarrow$	$A \sqsubseteq \widehat{D},\quad B \sqsubseteq \exists r.A$
NF4	$B \sqsubseteq D \sqcap E$	$\longrightarrow$	$B \sqsubseteq D,\quad B \sqsubseteq E$

where C, D, E denote arbitrary $\mathcal{EL}$ concepts,
$\widehat{D}, \widehat{E}$ denote $\mathcal{EL}$ concepts that are neither concept names nor $\top$,
B is a concept name, and
A is a *new* concept name.

Fig. 6.1. The normalisation rules for $\mathcal{EL}$.

On the right-hand side of each rule application, the GCIs that are not in normal form, and thus need to be further processed, are underlined. The concept names B_0, B_1, B_2, B_3 are new concept names that are introduced as "abbreviations" for compound concepts in the applications of the rules NF0, NF1$_\ell$, NF1$_r$ and NF2. Overall, the rule applications above transform the TBox $\mathcal{T} := \{\exists r.A \sqcap \exists r.\exists s.A \sqsubseteq A \sqcap B\}$ into the normalised TBox $\mathcal{T}' := \{\exists r.A \sqsubseteq B_1, B_1 \sqcap B_2 \sqsubseteq B_0, \exists s.A \sqsubseteq B_3, \exists r.B_3 \sqsubseteq B_2, B_0 \sqsubseteq A, B_0 \sqsubseteq B\}$.

Lemma 6.2. *Let $\mathcal{T}$ be a general $\mathcal{EL}$ TBox. Then $\mathcal{T}$ can be transformed into a normalised $\mathcal{EL}$ TBox $\mathcal{T}'$ by a linear number of applications of the rules of Figure 6.1. In addition, the size of the resulting TBox $\mathcal{T}'$ is linear in the size of $\mathcal{T}$.*[1]

Proof. We say that an occurrence of a concept $\widehat{D}$ within a general $\mathcal{EL}$ TBox is *abnormal* if one of the following conditions holds:

(i) $\widehat{D}$ is neither a concept name nor $\top$, and $\widehat{D}$ is the left-hand side of a GCI $\widehat{D} \sqsubseteq \widehat{E}$ whose right-hand side $\widehat{E}$ is neither a concept name nor $\top$;
(ii) $\widehat{D}$ is neither a concept name nor $\top$, and this occurrence is under a conjunction or an existential restriction operator;
(iii) this occurrence is under a conjunction operator on the right-hand side of a GCI.

[1] We use the definition of the size of a TBox as introduced in Chapter 3.

The *abnormality degree* of a general $\mathcal{EL}$ TBox is the number of abnormal occurrences of a concept in this TBox. Obviously, the abnormality degree of a TBox is bounded by the size of the TBox, and a TBox with abnormality degree 0 is normalised.

In a first phase of rule applications, we apply NF0 exhaustively. Each application of this rule decrements the abnormality degree by 1. In fact, the occurrence of the concept $\widehat{D}$ on the left-hand side of this rule is abnormal, while the occurrence of $\widehat{D}$ on the right-hand side is no longer abnormal. In addition, any abnormal occurrence of a concept within $\widehat{D}$ or $\widehat{E}$ in the new GCIs was also an abnormal occurrence in the old GCI. Thus, this first phase of rule applications stops after a linear number of steps. The resulting TBox contains only GCIs for which one of its sides is a concept name. Obviously, this property is preserved by applications of the other rules, which is the reason why on the left-hand sides of these rules we consider only GCIs satisfying this property.

In the second phase of rule applications, we apply the remaining rules exhaustively. Each application of such a rule decrements the abnormality degree by at least 1. For the rules $\mathsf{NF1}_r$, $\mathsf{NF1}_\ell$, NF2 and NF3, the occurrence of the concept $\widehat{D}$ on the left-hand side of these rules is abnormal, while the occurrence of $\widehat{D}$ on the right-hand side is no longer abnormal. In addition, no new abnormal occurrences of concepts are introduced by the rule application. For NF4, the occurrences of D and E are abnormal, and cease to be so after the rule is applied. Note that, because the left-hand side B of the GCI is a concept name, this left-hand side does not contain any abnormal occurrences of concepts, and thus the fact that the left-hand side is copied is harmless. This shows that the second phase of rule applications also stops after a linear number of steps. To be more precise, the overall number of rule applications in the two phases is bounded by the size of $\mathcal{T}$ since each rule application decrements the abnormality degree by at least 1 and the abnormality degree of $\mathcal{T}$ is bounded by the size of $\mathcal{T}$. When both phases are finished, the resulting TBox $\mathcal{T}'$ is normalised since a non-normalised GCI that has a concept name as one of its sides would trigger the application of one of the rules $\mathsf{NF1}_r$, $\mathsf{NF1}_\ell$, NF2, NF3, NF4.

Regarding the size of $\mathcal{T}'$, we note that an application of a rule adds at most 2 to the size of the TBox. The rules $\mathsf{NF0}, \ldots, \mathsf{NF3}$ increment the size by exactly 2 since they add two occurrences of A. The rule NF4 removes one conjunction operator, but duplicates B. However, since B is a concept name, which has size 1, the overall size of the TBox actually stays the same. Since the number of rule applications is bounded by the size of $\mathcal{T}$ and each rule application increments the size of the TBox by at most 2, the size of $\mathcal{T}'$ is at most three times the size of the original TBox $\mathcal{T}$. □

It remains to show that the original TBox $\mathcal{T}$ and the normalised TBox $\mathcal{T}'$ obtained from $\mathcal{T}$ using the rules of Figure 6.1 are in an appropriate semantic relationship that ensures that classification of the normalised TBox $\mathcal{T}'$ also yields the subsumption hierarchy for the concept names occurring in $\mathcal{T}$. One might be tempted to claim that $\mathcal{T}$ and $\mathcal{T}'$ are equivalent in the sense that they have the same models. This is not the case, however, because the rules of Figure 6.1 introduce new concept names. Thus, we first need to define an appropriate extension of the notion of equivalence.

Definition 6.3. For a given general $\mathcal{EL}$ TBox $\mathcal{T}_0$, its *signature* $sig(\mathcal{T}_0)$ consists of the concept and role names occurring in the GCIs of $\mathcal{T}_0$. Given general $\mathcal{EL}$ TBoxes $\mathcal{T}_1$ and $\mathcal{T}_2$, we say that $\mathcal{T}_2$ is a *conservative extension* of $\mathcal{T}_1$ if

- $sig(\mathcal{T}_1) \subseteq sig(\mathcal{T}_2)$,
- every model of $\mathcal{T}_2$ is a model of $\mathcal{T}_1$, and
- for every model $\mathcal{I}_1$ of $\mathcal{T}_1$ there exists a model $\mathcal{I}_2$ of $\mathcal{T}_2$ such that the extensions of concept and role names from $sig(\mathcal{T}_1)$ coincide in $\mathcal{I}_1$ and $\mathcal{I}_2$, i.e.,

  $A^{\mathcal{I}_1} = A^{\mathcal{I}_2}$ for all concept names $A \in sig(\mathcal{T}_1)$, and
  $r^{\mathcal{I}_1} = r^{\mathcal{I}_2}$ for all role names $r \in sig(\mathcal{T}_1)$.

It is easy to see that the notion of a conservative extension is *transitive*, i.e., if $\mathcal{T}_2$ is a conservative extension of $\mathcal{T}_1$ and $\mathcal{T}_3$ is a conservative extension of $\mathcal{T}_2$, then $\mathcal{T}_3$ is a conservative extension of $\mathcal{T}_1$.

In addition, the notion *preserves subsumption* in the following sense. If $\mathcal{T}_2$ is a conservative extension of $\mathcal{T}_1$, then subsumption with respect to $\mathcal{T}_1$ coincides with subsumption with respect to $\mathcal{T}_2$ for all concepts built using only symbols from $sig(\mathcal{T}_1)$.

Lemma 6.4. *Let $\mathcal{T}_1$ and $\mathcal{T}_2$ be general $\mathcal{EL}$ TBoxes such that $\mathcal{T}_2$ is a conservative extension of $\mathcal{T}_1$, and C, D are $\mathcal{EL}$ concepts containing only concept and role names from $sig(\mathcal{T}_1)$. Then $\mathcal{T}_1 \models C \sqsubseteq D$ if and only if $\mathcal{T}_2 \models C \sqsubseteq D$.*

Proof. First, assume that $\mathcal{T}_2 \not\models C \sqsubseteq D$. Then there is a model $\mathcal{I}$ of $\mathcal{T}_2$ such that $C^{\mathcal{I}} \not\subseteq D^{\mathcal{I}}$. Since $\mathcal{I}$ is also a model of $\mathcal{T}_1$, this implies $\mathcal{T}_1 \not\models C \sqsubseteq D$.

Second, assume that $\mathcal{T}_1 \not\models C \sqsubseteq D$. Then there is a model $\mathcal{I}_1$ of $\mathcal{T}_1$ such that $C^{\mathcal{I}_1} \not\subseteq D^{\mathcal{I}_1}$. Let $\mathcal{I}_2$ be a model of $\mathcal{T}_2$ such that the extensions of concept and role names from $sig(\mathcal{T}_1)$ coincide in $\mathcal{I}_1$ and $\mathcal{I}_2$. Since C, D contain only concept and role names from $sig(\mathcal{T}_1)$, we have $C^{\mathcal{I}_2} = C^{\mathcal{I}_1} \not\subseteq D^{\mathcal{I}_1} = D^{\mathcal{I}_2}$, and thus $\mathcal{T}_2 \not\models C \sqsubseteq D$. □

Because of this lemma, it is enough to show that the rules of Figure 6.1 transform a given TBox into a conservative extension of this TBox.

Proposition 6.5. *Assume that $\mathcal{T}_2$ is obtained from $\mathcal{T}_1$ by applying one of the rules of Figure 6.1. Then $\mathcal{T}_2$ is a conservative extension of $\mathcal{T}_1$.*

Proof. We treat the rule $\mathsf{NF1}_r$ in detail. The rules $\mathsf{NF0}$, $\mathsf{NF1}_\ell$, $\mathsf{NF2}$ and $\mathsf{NF3}$ can be treated similarly. The proposition holds trivially for $\mathsf{NF4}$ since in that case $\mathcal{T}_1$ and $\mathcal{T}_2$ have the same signature and are obviously equivalent.

Regarding $\mathsf{NF1}_r$, assume that $\mathcal{T}_2$ is obtained from $\mathcal{T}_1$ by replacing the GCI $C \sqcap \widehat{D} \sqsubseteq B$ with the two GCI $\widehat{D} \sqsubseteq A$ and $C \sqcap A \sqsubseteq B$, where A is a new concept name, i.e., $A \notin sig(\mathcal{T}_1)$. Obviously, $sig(\mathcal{T}_2) = sig(\mathcal{T}_1) \cup \{A\}$, and thus $sig(\mathcal{T}_1) \subseteq sig(\mathcal{T}_2)$. Next, assume that $\mathcal{I}$ is a model of $\mathcal{T}_2$. Then we have $\widehat{D}^{\mathcal{I}} \subseteq A^{\mathcal{I}}$ and $C^{\mathcal{I}} \cap A^{\mathcal{I}} \subseteq B^{\mathcal{I}}$. Obviously, this implies $C^{\mathcal{I}} \cap \widehat{D}^{\mathcal{I}} \subseteq C^{\mathcal{I}} \cap A^{\mathcal{I}} \subseteq B^{\mathcal{I}}$, and thus $\mathcal{I}$ is also a model of $\mathcal{T}_1$. Finally, assume that $\mathcal{I}_1$ is a model of $\mathcal{T}_1$. Let $\mathcal{I}_2$ be the interpretation that coincides with $\mathcal{I}_1$ on all concept and role names with the exception of A. For A, we define the extension in $\mathcal{I}_2$ as $A^{\mathcal{I}_2} := \widehat{D}^{\mathcal{I}_1}$. Since $\mathcal{I}_1$ is a model of $\mathcal{T}_1$, we have $C^{\mathcal{I}_1} \cap \widehat{D}^{\mathcal{I}_1} \subseteq B^{\mathcal{I}_1}$. In addition, since A does not occur in C, $\widehat{D}$ and B, we have $C^{\mathcal{I}_1} = C^{\mathcal{I}_2}$, $\widehat{D}^{\mathcal{I}_1} = \widehat{D}^{\mathcal{I}_2}$ and $B^{\mathcal{I}_1} = B^{\mathcal{I}_2}$. This yields $\widehat{D}^{\mathcal{I}_2} = \widehat{D}^{\mathcal{I}_1} = A^{\mathcal{I}_2}$ and $C^{\mathcal{I}_2} \cap A^{\mathcal{I}_2} = C^{\mathcal{I}_1} \cap \widehat{D}^{\mathcal{I}_1} \subseteq B^{\mathcal{I}_1} = B^{\mathcal{I}_2}$, which shows that $\mathcal{I}_2$ is a model of $\mathcal{T}_2$. □

Because of transitivity, the following corollary is an immediate consequence of this proposition and Lemma 6.4.

Corollary 6.6. *Let $\mathcal{T}$ be a general $\mathcal{EL}$ TBox and $\mathcal{T}'$ the normalised TBox obtained from $\mathcal{T}$ using the rules of Figure 6.1, as described in the proof of Lemma 6.2. Then we have*

$$\mathcal{T} \models A \sqsubseteq B \quad \textit{if and only if} \quad \mathcal{T}' \models A \sqsubseteq B$$

for all concept names $A, B \in sig(\mathcal{T})$.

$$\text{CR1}\quad \frac{}{A \sqsubseteq A} \qquad\qquad \text{CR2}\quad \frac{}{A \sqsubseteq \top}$$

$$\text{CR3}\quad \frac{A_1 \sqsubseteq A_2 \quad A_2 \sqsubseteq A_3}{A_1 \sqsubseteq A_3} \qquad\qquad \text{CR4}\quad \frac{A \sqsubseteq A_1 \quad A \sqsubseteq A_2 \quad A_1 \sqcap A_2 \sqsubseteq B}{A \sqsubseteq B}$$

$$\text{CR5}\quad \frac{A \sqsubseteq \exists r.A_1 \quad A_1 \sqsubseteq B_1 \quad \exists r.B_1 \sqsubseteq B}{A \sqsubseteq B}$$

Fig. 6.2. The classification rules for $\mathcal{EL}$.

6.1.2 The classification procedure

Let $\mathcal{T}$ be a general $\mathcal{EL}$ TBox in normal form. We start with the GCIs in $\mathcal{T}$ and add implied GCIs using appropriate inference rules. All the GCIs generated in this way are of a specific form.

Definition 6.7. A $\mathcal{T}$-*sequent* is a GCI of the form

$$A \sqsubseteq B, \quad A_1 \sqcap A_2 \sqsubseteq B, \quad A \sqsubseteq \exists r.B \ \text{ or } \ \exists r.A \sqsubseteq B,$$

where A, A_1, A_2, B are concept names in $sig(\mathcal{T})$ or the top concept $\top$, and r is a role name in $sig(\mathcal{T})$.

Obviously, the overall number of $\mathcal{T}$-sequents is polynomial in the size of $\mathcal{T}$, and every GCI in $\mathcal{T}$ is a $\mathcal{T}$-sequent. A set of $\mathcal{T}$-sequents consists of GCIs, and thus is a TBox. Inspired by its use in sequent calculi, we employ the name *sequent* rather than GCI to emphasise the fact that new $\mathcal{T}$-sequents can be derived using inference rules. The prefix $\mathcal{T}$ specifies the original TBox and restricts $\mathcal{T}$-sequents to being normalised GCIs containing only concept and role names from $sig(\mathcal{T})$.

Given the normalised input TBox $\mathcal{T}$, we define the current TBox $\mathcal{T}'$ to be initially $\mathcal{T}$, and then add new $\mathcal{T}$-sequents to $\mathcal{T}'$ by applying the *classification rules* of Figure 6.2. The rules given in this figure are, of course, not concrete rules, but rule schemata. To build a concrete instance of such a rule schema, the meta-variables A, A_1, A_2, B, B_1 must be replaced by a concrete $\mathcal{EL}$ concept and the meta-variable r by a concrete role name. However, it is important to note that *only instantiations are allowed for which all the GCIs occurring in the rule are $\mathcal{T}$-sequents.* A rule instance obtained in this way is then to be read as follows: if all the $\mathcal{T}$-sequents above the line occur in the current TBox $\mathcal{T}'$, then add the $\mathcal{T}$-sequent below the line to $\mathcal{T}'$ unless it already belongs to $\mathcal{T}'$. To simplify notation, we will in the following dispense with drawing a strict distinction between rule schemata and rule instances, and talk about applying a rule of Figure 6.2 rather than saying that we apply an instance of a rule schema.

Example 6.8. As an example, consider the TBox

$$\mathcal{T}_1 = \{A \sqsubseteq \exists r.A, \exists r.B \sqsubseteq B_1, \top \sqsubseteq B, A \sqsubseteq B_2, B_1 \sqcap B_2 \sqsubseteq C\}.$$

The rule CR2 can generate the $\mathcal{T}_1$-sequent $A \sqsubseteq \top$. Together with $\top \sqsubseteq B \in \mathcal{T}_1$, this $\mathcal{T}_1$-sequent can be used by rule CR3 to derive $A \sqsubseteq B$. This $\mathcal{T}_1$-sequent, together with the first and the second GCI in $\mathcal{T}_1$, can now be used by rule CR4 to infer $A \sqsubseteq B_1$. Finally, this $\mathcal{T}_1$-sequent, together with the third and the fourth GCI in $\mathcal{T}_1$, yields $A \sqsubseteq C$ by an application of rule CR5.

As a second example, consider the TBox

$$\mathcal{T}_2 = \{A \sqsubseteq \exists r.A, \exists r.A \sqsubseteq B\}.$$

Then there are two ways of deriving $A \sqsubseteq B$. One is by a direct application of rule CR3. The other is by first applying CR1 to derive $A \sqsubseteq A$, and then applying rule CR5.

The TBox obtained by an exhaustive application of the rules of Figure 6.2 to an initial normalised TBox $\mathcal{T}$ is denoted by $\mathcal{T}^*$. We call this process *saturation* of $\mathcal{T}$ with respect to the inference rules of Figure 6.2, and the resulting TBox $\mathcal{T}^*$ the *saturated* TBox. We will show that, for all concept names A, B (where $A, B \in sig(\mathcal{T}) \cup \{\top\}$), we then have

$$\mathcal{T} \models A \sqsubseteq B \text{ if and only if } A \sqsubseteq B \in \mathcal{T}^*. \tag{6.1}$$

But first note that the saturated TBox $\mathcal{T}^*$ is uniquely determined and can be computed in polynomial time.

Lemma 6.9. *The saturated TBox $\mathcal{T}^*$ is uniquely determined by $\mathcal{T}$, and it can be computed by a polynomial number of applications of the inference rules of Figure 6.2.*

Proof. Each rule application adds one new $\mathcal{T}$-sequent to $\mathcal{T}'$, and there are only polynomially many $\mathcal{T}$-sequents. Thus, after a polynomial number of rule applications, no new sequents can be added by the rules, and thus the application of rules terminates.

The choice of which applicable rule to apply during the saturation process does not influence the resulting TBox $\mathcal{T}^*$. Indeed, note that $\mathcal{T}$-sequents may be added to, but are never removed from, the TBox $\mathcal{T}'$. Thus, if the condition that the $\mathcal{T}$-sequents above the line of a rule occur in the current TBox $\mathcal{T}'$ is satisfied at some stage of the saturation process, then it remains satisfied also at later stages. Consequently, each applicable rule remains applicable until its consequent (i.e., the $\mathcal{T}$-sequent below the line) is added to $\mathcal{T}'$. □

Let us now show the "if" direction of (6.1). Obviously, this direction is an immediate consequence of the next lemma and the fact that any GCI in $\mathcal{T}$ follows from $\mathcal{T}$.

Lemma 6.10 (Soundness). *If all the GCIs in $\mathcal{T}'$ follow from $\mathcal{T}$ and the $\mathcal{T}$-sequents above the line of one of the inference rules of Figure 6.2 belong to $\mathcal{T}'$, then the $\mathcal{T}$-sequent below the line also follows from $\mathcal{T}$.*

Proof. This is an immediate consequence of the following facts:

- the subsumption relation $\sqsubseteq_\mathcal{T}$ is reflexive and transitive;
- $\top$ subsumes every concept with respect to any TBox;
- $A \sqsubseteq_\mathcal{T} A_1$ and $A \sqsubseteq_\mathcal{T} A_2$ implies $A \sqsubseteq_\mathcal{T} A_1 \sqcap A_2$;
- $A_1 \sqsubseteq_\mathcal{T} A_2$ implies $\exists r.A_1 \sqsubseteq_\mathcal{T} \exists r.A_2$.

Some of these facts have already been shown in Chapter 2. All of them are easy consequences of the semantics of the concept constructors of $\mathcal{EL}$ and the definition of subsumption. □

Instead of showing the "only if" direction of (6.1) directly, we prove its contrapositive, i.e., if $A \sqsubseteq B \notin \mathcal{T}^*$ then $\mathcal{T} \not\models A \sqsubseteq B$. For this purpose, we construct a model of $\mathcal{T}$ that does not satisfy the GCI $A \sqsubseteq B$.

Definition 6.11. Let $\mathcal{T}$ be a general $\mathcal{EL}$ TBox in normal form and $\mathcal{T}^*$ the saturated TBox obtained by exhaustive application of the inference rules of Figure 6.2. The *canonical interpretation* $\mathcal{I}_{\mathcal{T}^*}$ induced by $\mathcal{T}^*$ is defined as follows:

$$
\begin{aligned}
\Delta^{\mathcal{I}_{\mathcal{T}^*}} &= \{A \mid A \text{ is a concept name in } \mathit{sig}(\mathcal{T})\} \cup \{\top\},\\
A^{\mathcal{I}_{\mathcal{T}^*}} &= \{B \in \Delta^{\mathcal{I}_{\mathcal{T}^*}} \mid B \sqsubseteq A \in \mathcal{T}^*\} \text{ for all concept names } A \in \mathit{sig}(\mathcal{T}),\\
r^{\mathcal{I}_{\mathcal{T}^*}} &= \{(A,B) \in \Delta^{\mathcal{I}_{\mathcal{T}^*}} \times \Delta^{\mathcal{I}_{\mathcal{T}^*}} \mid A \sqsubseteq \exists r.B \in \mathcal{T}^*\}\\
&\qquad \text{for all role names } r \in \mathit{sig}(\mathcal{T}).
\end{aligned}
$$

Note that, according to this definition, we have $B \in A^{\mathcal{I}_{\mathcal{T}^*}}$ if and only if $B \sqsubseteq A \in \mathcal{T}^*$ for all concept names $A \in \mathit{sig}(\mathcal{T})$. The same is actually true for $A = \top$. In fact, $\top^{\mathcal{I}_{\mathcal{T}^*}} = \Delta^{\mathcal{I}_{\mathcal{T}^*}}$ according to the semantics of the top concept. Due to the presence of the rule $\mathsf{CR2}$, $B \sqsubseteq \top \in \mathcal{T}^*$ for all $B \in \Delta^{\mathcal{I}_{\mathcal{T}^*}}$.

Lemma 6.12. *The canonical interpretation induced by $\mathcal{T}^*$ is a model of the saturated TBox $\mathcal{T}^*$.*

Proof. All the GCIs in $\mathcal{T}^*$ are $\mathcal{T}$-sequents, i.e., they are of the form described in Definition 6.7.

- Consider a GCI of the form $A \sqsubseteq B \in \mathcal{T}^*$. If $A' \in A^{\mathcal{I}_{\mathcal{T}^*}}$, then we have $A' \sqsubseteq A \in \mathcal{T}^*$. Since $\mathcal{T}^*$ is saturated, the rule CR3 is no longer applicable, and thus we must have $A' \sqsubseteq B \in \mathcal{T}^*$. This yields $A' \in B^{\mathcal{I}_{\mathcal{T}^*}}$, and thus shows that $\mathcal{I}_{\mathcal{T}^*}$ satisfies the GCI $A \sqsubseteq B$.
- GCIs of the form $A_1 \sqcap A_2 \sqsubseteq B$ can be treated analogously, using the semantics of conjunction and the rule CR4 instead of CR3.
- Consider a GCI of the form $A \sqsubseteq \exists r.B \in \mathcal{T}^*$. If $A' \in A^{\mathcal{I}_{\mathcal{T}^*}}$, then we have $A' \sqsubseteq A \in \mathcal{T}^*$, and thus (due to CR3) $A' \sqsubseteq \exists r.B \in \mathcal{T}^*$. The definition of the interpretation of roles in $\mathcal{I}_{\mathcal{T}^*}$ thus yields $(A', B) \in r^{\mathcal{I}_{\mathcal{T}^*}}$. Finally, due to rule CR1, $B \sqsubseteq B \in \mathcal{T}^*$, and thus $B \in B^{\mathcal{I}_{\mathcal{T}^*}}$. This shows that $A' \in (\exists r.B)^{\mathcal{I}_{\mathcal{T}^*}}$.
- Consider a GCI of the form $\exists r.A \sqsubseteq B \in \mathcal{T}^*$. If $A' \in (\exists r.A)^{\mathcal{I}_{\mathcal{T}^*}}$, then there is $B' \in \Delta^{\mathcal{I}_{\mathcal{T}^*}}$ such that $(A', B') \in r^{\mathcal{I}_{\mathcal{T}^*}}$ and $B' \in A^{\mathcal{I}_{\mathcal{T}^*}}$. This yields $A' \sqsubseteq \exists r.B' \in \mathcal{T}^*$ and $B' \sqsubseteq A \in \mathcal{T}^*$. Thus, due to rule CR5, $A' \sqsubseteq B \in \mathcal{T}^*$, which yields $A' \in B^{\mathcal{I}_{\mathcal{T}^*}}$.

Since all the GCIs in $\mathcal{T}^*$ are of one of the forms considered above, we have thus shown that $\mathcal{I}_{\mathcal{T}^*}$ does indeed satisfy every GCI in $\mathcal{T}^*$. □

The "only if" direction of (6.1) is an easy consequence of this lemma.

Lemma 6.13 (Completeness)**.** *Let $\mathcal{T}$ be a general $\mathcal{EL}$ TBox in normal form and $\mathcal{T}^*$ the saturated TBox obtained by exhaustive application of the inference rules of Figure 6.2. Then $\mathcal{T} \models A \sqsubseteq B$ implies $A \sqsubseteq B \in \mathcal{T}^*$.*

Proof. As mentioned above, we show the contrapositive of the statement of the lemma. Thus, assume that $A \sqsubseteq B \notin \mathcal{T}^*$. Then $A \notin B^{\mathcal{I}_{\mathcal{T}^*}}$ by the definition of the interpretation of concept names in $\mathcal{I}_{\mathcal{T}^*}$. Due to CR1, we have $A \sqsubseteq A \in \mathcal{T}^*$, and thus $A \in A^{\mathcal{I}_{\mathcal{T}^*}}$. This shows that $\mathcal{I}_{\mathcal{T}^*}$ does not satisfy the GCI $A \sqsubseteq B$. Since $\mathcal{I}_{\mathcal{T}^*}$ is a model of the saturated TBox $\mathcal{T}^*$, it is also a model of its subset $\mathcal{T}$, which yields $\mathcal{T} \not\models A \sqsubseteq B$. □

If we put all the results of this section together, we obtain the following theorem.

Theorem 6.14. *Subsumption in $\mathcal{EL}$ with respect to general TBoxes is decidable in polynomial time.*

Proof. Let $\mathcal{T}_0$ be a general $\mathcal{EL}$ TBox and C, D $\mathcal{EL}$ concepts. To decide whether $\mathcal{T}_0 \models C \sqsubseteq D$ holds or not, we first add the GCIs $A \sqsubseteq C, D \sqsubseteq B$ to $\mathcal{T}_0$. The resulting TBox $\mathcal{T}_1$ is then normalised using the normalisation rules of Figure 6.1, as described in the proof of Lemma 6.2. The size of the normalised TBox $\mathcal{T}$ obtained this way is linear in the size of $\mathcal{T}_0$, and we have $\mathcal{T}_0 \models C \sqsubseteq D$ if and only if $\mathcal{T} \models A \sqsubseteq B$.

Let $\mathcal{T}^*$ be the TBox obtained by an exhaustive application of the rules of Figure 6.2, starting with $\mathcal{T}$. We know that the saturation process requires only a polynomial number of rule applications. Since a single rule application can be done in polynomial time, this shows that $\mathcal{T}^*$ can be computed in time polynomial in the size of $\mathcal{T}$, and thus also in the size of $\mathcal{T}_0$. In addition, we have $\mathcal{T} \models A \sqsubseteq B$ if and only if $A \sqsubseteq B \in \mathcal{T}^*$. Thus, by checking whether $A \sqsubseteq B$ is an element of $\mathcal{T}^*$, we can decide whether $\mathcal{T}_0 \models C \sqsubseteq D$ holds or not. □

6.2 Subsumption in $\mathcal{ELI}$

In this section, we show that the ideas underlying the subsumption algorithm of the previous section can also be used to obtain a subsumption algorithm for $\mathcal{ELI}$, the extension of $\mathcal{EL}$ by inverse roles. However, as mentioned in the introduction to this chapter, subsumption in $\mathcal{ELI}$ is no longer polynomial, but ExpTime-complete. One reason for the higher complexity of subsumption in $\mathcal{ELI}$ is that it can express a restricted form of value restrictions, and thus comes close to $\mathcal{FL}_0$. In fact, it is easy to see that the GCI $\exists r^-.C \sqsubseteq D$ is equivalent to the GCI $C \sqsubseteq \forall r.D$. Thus, $\mathcal{ELI}$ can express value restrictions on the right-hand side of GCIs (but not on the left).

As usual, we will use r^- to denote s if $r = s^-$ for a role name s.

6.2.1 Normalisation

In principle, $\mathcal{ELI}$ admits a normal form that is similar to the one for $\mathcal{EL}$ introduced above. The only differences are that inverse roles can occur in place of role names and that we rewrite each GCIs of the form $\exists r^-.A \sqsubseteq B$ into the equivalent GCI $A \sqsubseteq \forall r.B$, where r is a role name or the inverse of a role name. To be more precise, we say that the general $\mathcal{ELI}$ TBox $\mathcal{T}$ is *in i.normal form* (or is *i.normalised*) if all its GCI are of one of the following forms:

$$A \sqsubseteq B, \quad A_1 \sqcap A_2 \sqsubseteq B, \quad A \sqsubseteq \exists r.B \ \text{ or } \ A \sqsubseteq \forall r.B,$$

where A, A_1, A_2, B are concept names or the top concept $\top$, and r is a role name or the inverse of a role name. The normalisation rules for $\mathcal{EL}$, extended by a rule that rewrites GCIs with existential restrictions on the left-hand side into the equivalent ones with value restrictions on the right-hand side, can be used to generate this i.normal form.

Corollary 6.15. *Given a general $\mathcal{ELI}$ TBox $\mathcal{T}$, we can compute in polynomial time an i.normalised $\mathcal{ELI}$ TBox $\mathcal{T}'$ that is a conservative extension of $\mathcal{T}$. In particular, we have*

$$\mathcal{T} \models A \sqsubseteq B \;\; \textit{if and only if} \;\; \mathcal{T}' \models A \sqsubseteq B$$

for all concept names $A, B \in sig(\mathcal{T})$.

6.2.2 The classification procedure

In the following, we assume that $\mathcal{T}$ is a general $\mathcal{ELI}$ TBox in i.normal form. The higher complexity of subsumption in $\mathcal{ELI}$ necessitates the use of an extended notion of sequents within our classification procedure.

Definition 6.16. A $\mathcal{T}$-*i.sequent* is an expression of the form

$$K \sqsubseteq \{A\}, \quad K \sqsubseteq \exists r.K' \;\text{ or }\; K \sqsubseteq \forall r.\{A\},$$

where K, K' are *sets* of concept names in $sig(\mathcal{T})$, A is a concept name in $sig(\mathcal{T})$ and r is a role name in $sig(\mathcal{T})$ or the inverse of a role name in $sig(\mathcal{T})$.

From a semantic point of view, a set in a $\mathcal{T}$-i.sequent stands for the conjunction of its elements, where the empty conjunction corresponds to $\top$. Consequently, $\mathcal{T}$-i.sequents are GCIs, and thus a set of $\mathcal{T}$-i.sequents is a general $\mathcal{ELI}$ TBox. Obviously, the overall number of $\mathcal{T}$-i.sequents is exponential in the size of $\mathcal{T}$. In addition, every GCI in the i.normalised TBox $\mathcal{T}$ is either equivalent to a $\mathcal{T}$-i.sequent or a tautology, i.e., satisfied in every interpretation. In the first case, we respresent it as a $\mathcal{T}$-i.sequent, and in the second case, we remove it. For example, the GCI $\top \sqsubseteq A$ corresponds to the $\mathcal{T}$-i.sequent $\emptyset \sqsubseteq \{A\}$, and the GCI $A_1 \sqcap A_2 \sqsubseteq B$ corresponds to the $\mathcal{T}$-i.sequent $\{A_1, A_2\} \sqsubseteq \{B\}$. GCIs with $\top$ or $\forall r.\top$ on the right-hand side are obviously tautologies.

Given the i.normalised input TBox $\mathcal{T}$, we define the current TBox $\mathcal{T}'$ to consist initially of the non-tautological GCIs in $\mathcal{T}$ represented as $\mathcal{T}$-i.sequents. Then, we add new $\mathcal{T}$-i.sequents to $\mathcal{T}'$ by applying the *classification rules* of Figure 6.3.

 https://doi.org/10.1017/9781139025355.006

$$\text{i.CR1} \quad \frac{}{K \sqsubseteq \{A\}} \quad \text{if } A \in K \text{ and } K \text{ occurs in } \mathcal{T}'$$

$$\text{i.CR2} \quad \frac{M \sqsubseteq \{B\} \text{ for all } B \in K \quad K \sqsubseteq C}{M \sqsubseteq C} \quad \text{if } M \text{ occurs in } \mathcal{T}'$$

$$\text{i.CR3} \quad \frac{M_2 \sqsubseteq \exists r.M_1 \quad M_1 \sqsubseteq \forall r^-.\{A\}}{M_2 \sqsubseteq \{A\}}$$

$$\text{i.CR4} \quad \frac{M_1 \sqsubseteq \exists r.M_2 \quad M_1 \sqsubseteq \forall r.\{A\}}{M_1 \sqsubseteq \exists r.(M_2 \cup \{A\})}$$

Fig. 6.3. The classification rules for $\mathcal{ELI}$.

As in the previous section, the rules given in this figure are actually rule schemata. To build a concrete instance of such a rule schema, the meta-variables K, M, M_1, M_2 must be replaced by sets of concept names in $sig(\mathcal{T})$, the meta-variable A by a concept name in $sig(\mathcal{T})$ and the meta-variable r by a role name in $sig(\mathcal{T})$ or the inverse of a role name in $sig(\mathcal{T})$. The meta-variable C can be replaced by any expression that is an admissible right-hand side of a $\mathcal{T}$-i.sequent.

For the rule schema i.CR1, only instantiations are allowed for which the set of concept names K actually occurs explicitly in some $\mathcal{T}$-i.sequent in the current TBox $\mathcal{T}'$. The reason for this restriction is that without it the procedure would always generate an exponential number of $\mathcal{T}$-i.sequents, since there are exponentially many sets K of concept names in $sig(\mathcal{T})$. The analogous restriction on M in rule i.CR2 is needed in the case where $K = \emptyset$. In fact, in this case the condition "$M \sqsubseteq \{B\}$ for all $B \in K$" is trivially satisfied for all sets M of concept names in $sig(\mathcal{T})$. Thus, without the restriction, the presence of a $\mathcal{T}$-i.sequent of the form $\emptyset \sqsubseteq C$ would cause the generation of exponentially many $\mathcal{T}$-i.sequents of the form $M \sqsubseteq C$.

Though in general the generation of exponentially many $\mathcal{T}$-i.sequents cannot be avoided, the restriction on the applicability of rules i.CR1 and i.CR2 to sets K and M, respectively, already occurring in $\mathcal{T}'$, prevents such an explosion in cases where it is not needed.

Example 6.17. For example, if $\mathcal{T} = \{A \sqsubseteq B\} \cup \{A_i \sqsubseteq A_i \mid 1 \leq i \leq n\}$, then we have $\mathcal{T} \models M \cup \{A\} \sqsubseteq \{B\}$ for all (exponentially many) sets $\emptyset \neq M \subseteq \{A_1, \ldots, A_n\}$. However, due to the restriction on the applicability of rule i.CR1, none of these $\mathcal{T}$-i.sequents is actually generated by the calculus when applied to $\mathcal{T}' = \{\{A\} \sqsubseteq \{B\}\} \cup \{\{A_i\} \sqsubseteq \{A_i\} \mid 1 \leq i \leq n\}$. In fact, since none of the sets $M \cup \{A\}$ occurs in $\mathcal{T}'$, the rule i.CR1 is not applicable.

What may seem to be a completeness problem is in fact an important feature of the calculus, aiming to avoid a combinatorial explosion due to the derivation of exponentially many "uninteresting" consequences such as the ones in the above example. The next example shows in what situations the rules i.CR1 and i.CR2 are actually needed.

Example 6.18. If $\mathcal{T} = \{A \sqsubseteq \exists r.(A_1 \sqcap A_2 \sqcap A_3), \exists r.(A_1 \sqcap A_2) \sqsubseteq B\}$, then obviously $\mathcal{T} \models A \sqsubseteq B$. The set of $\mathcal{T}$-i.sequents corresponding to $\mathcal{T}$ is $\mathcal{T}' = \{\{A\} \sqsubseteq \exists r.\{A_1, A_2, A_3\}, \{A_1, A_2\} \sqsubseteq \forall r^-.\{B\}\}$. We show that the rules of Figure 6.3 can be used to derive the $\mathcal{T}$-i.sequent $\{A\} \sqsubseteq \{B\}$.

In fact, two applications of i.CR1 yield the $\mathcal{T}$-i.sequents $\{A_1, A_2, A_3\} \sqsubseteq \{A_1\}$ and $\{A_1, A_2, A_3\} \sqsubseteq \{A_2\}$. These applications are admissible since $\{A_1, A_2, A_3\}$ occurs in $\mathcal{T}'$. Given the two derived $\mathcal{T}$-i.sequents together with the second $\mathcal{T}$-i.sequent in $\mathcal{T}'$, an application of i.CR2 now yields $\{A_1, A_2, A_3\} \sqsubseteq \forall r^-.\{B\}$. Given this $\mathcal{T}$-i.sequent together with the first $\mathcal{T}$-i.sequent in $\mathcal{T}'$, an application of i.CR3 yields $\{A\} \sqsubseteq \{B\}$.

Due to the occurrence restrictions, the rules i.CR1 and i.CR2 cannot introduce new sets of concept names into $\mathcal{T}'$. The same is obviously true (without any restriction) for i.CR3. In contrast, rule i.CR4 can generate sets not yet occurring in $\mathcal{T}'$, and thus may cause an exponential blowup.

Example 6.19. Consider the $\mathcal{ELI}$ TBox $\mathcal{T} := \{A \sqsubseteq \exists r.\top\} \cup \{\exists r^-.A \sqsubseteq A_i \mid i = 1, \ldots, n\}$. By i.normalisation, we can transform this TBox into the following set of $\mathcal{T}$-i.sequents:

$$\mathcal{T}' := \{\{A\} \sqsubseteq \exists r.\emptyset\} \cup \{\{A\} \sqsubseteq \forall r.\{A_i\} \mid i = 1, \ldots, n\}.$$

It is easy to see that repeated applications of rule i.CR4 can now be used to generate all $\mathcal{T}$-i.sequents $\{A\} \sqsubseteq \exists r.M$ for $M \subseteq \{A_1, \ldots, A_n\}$.

Thus, if we add $M \sqsubseteq \forall r^-.\{B\}$ to $\mathcal{T}'$ for some set $M \subseteq \{A_1, \ldots, A_n\}$, then $\{A\} \sqsubseteq \{B\}$ can be derived by an application of i.CR3. Note, however, that for this it would have been sufficient to derive (by n applications of i.CR4) only the "maximal" $\mathcal{T}$-i.sequent $\{A\} \sqsubseteq \exists r.\{A_1, \ldots, A_n\}$, and then use a derivation of $\{A\} \sqsubseteq \{B\}$ analogous to the one shown in Example 6.18. It is thus imaginable that the exponential blowup demonstrated by this example could actually be avoided by a clever strategy. That this cannot always be the case follows from the fact that subsumption in $\mathcal{ELI}$ is ExpTime-complete. Later, in Section 6.3.1, we will give an example in which exponentially many $\mathcal{T}$-i.sequents need to be derived before the final $\mathcal{T}$-i.sequent $\{A\} \sqsubseteq \{B\}$ is reached.

Before analysing the complexity of the algorithm in more detail, we will show that it is actually sound and complete in the sense made precise in Proposition 6.20 below. Using the same notation as in the previous section, we denote the TBox obtained by an exhaustive application of the rules of Figure 6.3 as $\mathcal{T}^*$. We call this process *i.saturation* of $\mathcal{T}$ with respect to the inference rules of Figure 6.3, and the resulting TBox $\mathcal{T}^*$ the *i.saturated* TBox. As in the case of saturation for $\mathcal{EL}$, it is easy to see that the i.saturated TBox $\mathcal{T}^*$ is uniquely determined by $\mathcal{T}$.

Proposition 6.20. *For all concept names A, B in $sig(\mathcal{T})$ such that $\{A\}$ occurs in $\mathcal{T}^*$, we have $\mathcal{T} \models A \sqsubseteq B$ if and only if $\{A\} \sqsubseteq \{B\} \in \mathcal{T}^*$.*

Note that the condition "$\{A\}$ occurs in $\mathcal{T}^*$" can easily be satisfied for a given concept name A in $sig(\mathcal{T})$. For example, we can add the dummy GCI $A \sqsubseteq A$ to the input TBox, which is translated into the $\mathcal{T}$-i.sequent $\{A\} \sqsubseteq \{A\}$.

The "if" direction of this proposition is an immediate consequence of the next lemma and the fact that any GCI in $\mathcal{T}$ follows from $\mathcal{T}$.

Lemma 6.21 (Soundness)**.** *If all the GCIs in $\mathcal{T}'$ follow from $\mathcal{T}$ and the $\mathcal{T}$-i.sequents above the line of one of the inference rules of Figure 6.3 belong to $\mathcal{T}'$, then the $\mathcal{T}$-i.sequent below the line also follows from $\mathcal{T}$.*

Proof. Soundness of rule i.CR1 follows from the fact that a conjunction of concept names is subsumed by each of its conjuncts.

Soundness of rule i.CR2 is due to transitivity of subsumption and the fact that $\mathcal{T} \models M \sqsubseteq \{B\}$ for all $B \in K$ if and only if $\mathcal{T} \models M \sqsubseteq K$. Note that this fact is also true in the case where K is the empty set.

To see soundness of i.CR3, note that $\mathcal{T} \models M_1 \sqsubseteq \forall r^-.\{A\}$ if and only if $\mathcal{T} \models \exists r.M_1 \sqsubseteq \{A\}$. Thus transitivity of subsumption yields $\mathcal{T} \models M_2 \sqsubseteq \{A\}$.

Finally, to show soundness of rule i.CR4, assume that $\mathcal{I}$ is a model of $\mathcal{T}$. Thus, according to the assumptions in the formulation of the lemma, $\mathcal{I}$ satisfies the two GCIs above the line of rule i.CR4. We must show that it also satisfies the GCI below the line. To this end, consider an element $d \in M_1^{\mathcal{I}}$. By the first GCI above the line, there is an element $e \in \Delta^{\mathcal{I}}$ such that $(d, e) \in r^{\mathcal{I}}$ and $e \in M_2^{\mathcal{I}}$. Due to the second GCI above the line, we know that $d \in (\forall r.\{A\})^{\mathcal{I}}$, and thus $e \in A^{\mathcal{I}}$. Together with $e \in M_2^{\mathcal{I}}$, this yields $e \in (M_1 \cup \{A\})^{\mathcal{I}}$. Consequently, we have $d \in (\exists r.(M_2 \cup \{A\}))^{\mathcal{I}}$ as required. □

In order to show the "only if" direction of the proposition, we construct an appropriate canonical interpretation.

Definition 6.22 (Canonical interpretation)**.** Let $\mathcal{T}$ be a general $\mathcal{ELI}$ TBox in i.normal form and $\mathcal{T}^*$ the i.saturated TBox obtained by exhaustive application of the inference rules of Figure 6.3. The *canonical interpretation* $\mathcal{I}_{\mathcal{T}^*}$ induced by $\mathcal{T}^*$ is defined as follows:

$$
\begin{aligned}
\Delta^{\mathcal{I}_{\mathcal{T}^*}} &= \{M \mid M \text{ is a set of concept names in } \mathit{sig}(\mathcal{T}) \text{ that occurs in } \mathcal{T}^*\},\\
A^{\mathcal{I}_{\mathcal{T}^*}} &= \{M \in \Delta^{\mathcal{I}_{\mathcal{T}^*}} \mid M \sqsubseteq \{A\} \in \mathcal{T}^*\},\\
s^{\mathcal{I}_{\mathcal{T}^*}} &= \{(M,N) \in \Delta^{\mathcal{I}_{\mathcal{T}^*}} \times \Delta^{\mathcal{I}_{\mathcal{T}^*}} \mid M \sqsubseteq \exists s.N \in \mathcal{T}^* \text{ and } N \text{ is maximal},\\
&\qquad \text{i.e., there is no } N' \supsetneq N \text{ such that } M \sqsubseteq \exists s.N' \in \mathcal{T}^*\} \cup\\
&\quad\ \{(N,M) \in \Delta^{\mathcal{I}_{\mathcal{T}^*}} \times \Delta^{\mathcal{I}_{\mathcal{T}^*}} \mid M \sqsubseteq \exists s^-.N \in \mathcal{T}^* \text{ and } N \text{ is maximal},\\
&\qquad \text{i.e., there is no } N' \supsetneq N \text{ such that } M \sqsubseteq \exists s^-.N' \in \mathcal{T}^*\},
\end{aligned}
$$

where A ranges over all concept names in $\mathit{sig}(\mathcal{T})$ and s over all role names in $\mathit{sig}(\mathcal{T})$.

Our definition of the extension of role names in the canonical interpretation is symmetric with respect to the inverse operator, and thus also inverse roles satisfy the identity given in this definition.

Lemma 6.23. *Let r be a role name or the inverse of a role name. Then*

$$
\begin{aligned}
r^{\mathcal{I}_{\mathcal{T}^*}} = &\{(M,N) \in \Delta^{\mathcal{I}_{\mathcal{T}^*}} \times \Delta^{\mathcal{I}_{\mathcal{T}^*}} \mid M \sqsubseteq \exists r.N \in \mathcal{T}^*,\ N \textit{ maximal}\} \cup\\
&\{(N,M) \in \Delta^{\mathcal{I}_{\mathcal{T}^*}} \times \Delta^{\mathcal{I}_{\mathcal{T}^*}} \mid M \sqsubseteq \exists r^-.N \in \mathcal{T}^*,\ N \textit{ maximal}\}.
\end{aligned}
$$

Proof. If $r = s$ is a role name, then this identity is just the definition of $s^{\mathcal{I}_{\mathcal{T}^*}}$. Otherwise, if $r = s^-$ for a role name s, then this identity follows from the fact that $r^- = s$, the semantics of the inverse operator and the definition of $s^{\mathcal{I}_{\mathcal{T}^*}}$:

$$
\begin{aligned}
r^{\mathcal{I}_{\mathcal{T}^*}} = (s^-)^{\mathcal{I}_{\mathcal{T}^*}} &= \{(L,K) \in \Delta^{\mathcal{I}_{\mathcal{T}^*}} \times \Delta^{\mathcal{I}_{\mathcal{T}^*}} \mid (K,L) \in s^{\mathcal{I}_{\mathcal{T}^*}}\}\\
&= \{(N,M) \in \Delta^{\mathcal{I}_{\mathcal{T}^*}} \times \Delta^{\mathcal{I}_{\mathcal{T}^*}} \mid M \sqsubseteq \exists s.N \in \mathcal{T}^*,\ N \text{ maximal}\} \cup\\
&\quad\ \{(M,N) \in \Delta^{\mathcal{I}_{\mathcal{T}^*}} \times \Delta^{\mathcal{I}_{\mathcal{T}^*}} \mid M \sqsubseteq \exists s^-.N \in \mathcal{T}^*,\ N \text{ maximal}\}\\
&= \{(N,M) \in \Delta^{\mathcal{I}_{\mathcal{T}^*}} \times \Delta^{\mathcal{I}_{\mathcal{T}^*}} \mid M \sqsubseteq \exists r^-.N \in \mathcal{T}^*,\ N \text{ maximal}\} \cup\\
&\quad\ \{(M,N) \in \Delta^{\mathcal{I}_{\mathcal{T}^*}} \times \Delta^{\mathcal{I}_{\mathcal{T}^*}} \mid M \sqsubseteq \exists r.N \in \mathcal{T}^*,\ N \text{ maximal}\}.
\end{aligned}
$$

□

As in the case of $\mathcal{EL}$, it is now easy to show that the canonical interpretation is a model of the i.saturated TBox it is induced by.

Lemma 6.24. *The canonical interpretation induced by $\mathcal{T}^*$ is a model of the i.saturated TBox $\mathcal{T}^*$.*

Proof. All the GCIs in $\mathcal{T}^*$ are $\mathcal{T}$-i.sequents, i.e., they are of the form described in Definition 6.16.

- Consider a GCI of the form $K \sqsubseteq \{A\} \in \mathcal{T}^*$, and let $M \in K^{\mathcal{I}_{\mathcal{T}^*}}$, i.e., $M \sqsubseteq \{B\} \in \mathcal{T}^*$ for all $B \in K$. Then rule i.CR2 yields $M \sqsubseteq \{A\} \in \mathcal{T}^*$, and thus $M \in A^{\mathcal{I}_{\mathcal{T}^*}}$.
- Consider a GCI of the form $K \sqsubseteq \exists r.K' \in \mathcal{T}^*$ for a role name or the inverse of a role name r. Now assume that $M \in K^{\mathcal{I}_{\mathcal{T}^*}}$, i.e., $M \sqsubseteq \{B\} \in \mathcal{T}^*$ for all $B \in K$. Then rule i.CR2 yields $M \sqsubseteq \exists r.K' \in \mathcal{T}^*$, and thus there is a maximal set $K'' \supseteq K'$ with $M \sqsubseteq \exists r.K'' \in \mathcal{T}^*$ and $(M, K'') \in r^{\mathcal{I}_{\mathcal{T}^*}}$. Since K'' occurs in $\mathcal{T}^*$, rule i.CR1 yields $K'' \sqsubseteq \{A\} \in \mathcal{T}^*$ for all $A \in K''$. Since $K' \subseteq K''$, this implies $K'' \in K'^{\mathcal{I}_{\mathcal{T}^*}}$. Consequently, we have $M \in (\exists r.K')^{\mathcal{I}_{\mathcal{T}^*}}$.
- Consider a GCI of the form $K \sqsubseteq \forall r.\{A\} \in \mathcal{T}^*$ for a role name or the inverse of a role name r. Assume $M_1 \in K^{\mathcal{I}_{\mathcal{T}^*}}$ and that there is an M_2 such that $(M_1, M_2) \in r^{\mathcal{I}_{\mathcal{T}^*}}$. We must show that $M_2 \in A^{\mathcal{I}_{\mathcal{T}^*}}$.

  By the definition of $\mathcal{I}_{\mathcal{T}^*}$, $M_1 \in K^{\mathcal{I}_{\mathcal{T}^*}}$ yields $M_1 \sqsubseteq \{B\} \in \mathcal{T}^*$ for all $B \in K$. Because of rule i.CR2 we thus have $M_1 \sqsubseteq \forall r.\{A\} \in \mathcal{T}^*$.

  There are two possible reasons for (M_1, M_2) to belong to $r^{\mathcal{I}_{\mathcal{T}^*}}$.

  - First, assume that $M_1 \sqsubseteq \exists r.M_2 \in \mathcal{T}^*$ where M_2 is maximal with this property. Then rule i.CR4 yields $M_1 \sqsubseteq \exists r.(M_2 \cup \{A\}) \in \mathcal{T}^*$, and thus $A \in M_2$ due to the maximality of M_2. Since M_2 occurs in $\mathcal{T}^*$, rule i.CR1 yields $M_2 \sqsubseteq \{A\} \in \mathcal{T}^*$, and thus $M_2 \in A^{\mathcal{I}_{\mathcal{T}^*}}$ as required.
  - Second, assume that $M_2 \sqsubseteq \exists r^-.M_1 \in \mathcal{T}^*$, where M_1 is maximal with this property. Then rule i.CR3 yields $M_2 \sqsubseteq \{A\} \in \mathcal{T}^*$, and thus again $M_2 \in A^{\mathcal{I}_{\mathcal{T}^*}}$ as required.

Since all the elements of $\mathcal{T}^*$ are of one of the forms considered above, this shows that $\mathcal{I}_{\mathcal{T}^*}$ is indeed a model of $\mathcal{T}^*$. □

The first case (i.e., where $M_1 \sqsubseteq \exists r.M_2 \in \mathcal{T}^*$) in the treatment of value restrictions in the above proof makes clear why we need the maximality condition in the definition of the extensions of roles in the canonical model. Let us illustrate this issue using Example 6.19. There, we obtain all the $\mathcal{T}$-i.sequents $\{A\} \sqsubseteq \exists r.M$ for $M \subseteq \{A_1, \ldots, A_n\}$. Thus, the set $\{A\}$ and all the sets $M \subseteq \{A_1, \ldots, A_n\}$ are in the domain of the canonical model. However, only the pair $(\{A\}, \{A_1, \ldots, A_n\})$ belongs to the interpretation of r. In fact, adding any other pair $(\{A\}, M)$ with $M \subset \{A_1, \ldots, A_n\}$ would violate one of the GCIs $\{A\} \sqsubseteq \forall r.\{A_i\}$. To be more precise, assume that $A_i \notin M$. Then $M \sqsubseteq \{A_i\}$ cannot be derived, and thus M does not belong to the extension of A_i in the canonical model.

Given Lemma 6.24, completeness is now easy to show.

Lemma 6.25 (Completeness)**.** *Let A, B in $sig(\mathcal{T})$ be such that $\{A\}$ occurs in $\mathcal{T}^*$. Then $\mathcal{T} \models A \sqsubseteq B$ implies $\{A\} \sqsubseteq \{B\} \in \mathcal{T}^*$.*

Proof. We show the contrapositive. Assume that $\{A\} \sqsubseteq \{B\} \notin \mathcal{T}^*$. Since $\{A\}$ occurs in $\mathcal{T}^*$, we have $\{A\} \in \Delta^{\mathcal{I}_{\mathcal{T}^*}}$. Rule i.CR1 yields $\{A\} \sqsubseteq \{A\} \in \mathcal{T}^*$, and thus $\{A\} \in A^{\mathcal{I}_{\mathcal{T}^*}}$. However, $\{A\} \sqsubseteq \{B\} \notin \mathcal{T}^*$ shows that $\{A\} \notin B^{\mathcal{I}_{\mathcal{T}^*}}$. Since $\mathcal{I}_{\mathcal{T}^*}$ is a model of $\mathcal{T}^*$, and thus also of $\mathcal{T}$, this yields $\mathcal{T} \not\models A \sqsubseteq B$. □

If we put all the results of this section together, we obtain the following theorem.

Theorem 6.26. *Subsumption in $\mathcal{ELI}$ with respect to general TBoxes is decidable in exponential time.*

Proof. Let $\mathcal{T}_0$ be a general $\mathcal{ELI}$ TBox and C, D $\mathcal{ELI}$ concepts. To decide whether $\mathcal{T}_0 \models C \sqsubseteq D$ holds or not, we first add the GCIs $A \sqsubseteq C, D \sqsubseteq B$ to $\mathcal{T}_0$. The resulting TBox $\mathcal{T}_1$ is then i.normalised using the normalisation rules of Figure 6.1 together with the rule that transforms a GCI with an existential restriction on the left-hand side into the equivalent one with a value restriction on the right-hand side. The size of the i.normalised TBox $\mathcal{T}$ obtained this way is linear in the size of $\mathcal{T}_0$, and we have $\mathcal{T}_0 \models C \sqsubseteq D$ if and only if $\mathcal{T} \models A \sqsubseteq B$.

Let $\mathcal{T}^*$ be the TBox obtained by an exhaustive application of the rules of Figure 6.3, starting with $\mathcal{T}'$, in which the non-tautological GCIs in $\mathcal{T}$ are represented as $\mathcal{T}$-i.sequents. The i.saturated TBox $\mathcal{T}^*$ can be computed in time exponential in the size of $\mathcal{T}$ (and thus also in the size of $\mathcal{T}_0$), since there are only exponentially many $\mathcal{T}$-i.sequents and every application of a rule adds a $\mathcal{T}$-i.sequent. Since $\mathcal{T}_0$ contains a GCI whose left-hand side is A, the initial set of $\mathcal{T}$-i.sequents $\mathcal{T}'$ contains the set $\{A\}$. Thus, Lemma 6.25 yields $\mathcal{T} \models A \sqsubseteq B$ if and only if $\{A\} \sqsubseteq \{B\} \in \mathcal{T}^*$. Consequently, by checking whether $\{A\} \sqsubseteq \{B\}$ is an element of $\mathcal{T}^*$, we can decide whether $\mathcal{T}_0 \models C \sqsubseteq D$ holds or not. □

6.3 Comparing the two subsumption algorithms

First, we compare the two algorithms on the technical level of the rule sets, and then we take a more abstract point of view.

6.3.1 Comparing the classification rules

In principle, the classification rules for $\mathcal{ELI}$ are a generalisation of the rules for $\mathcal{EL}$, though at first sight the rules given in each of the two previous sections may look quite different from each other. In the following, we explain the connection between the two rule sets.

Obviously, rule $\mathsf{CR1}$ is the special case of rule $\mathsf{i.CR1}$ where $K = \{A\}$. The generalisation of rule $\mathsf{CR1}$ to $\mathsf{i.CR1}$ is needed to deal with the generalised form of sequents containing sets of concept names.

Rule $\mathsf{CR2}$ does not have a corresponding rule in the calculus for $\mathcal{ELI}$. Basically, the reason for this is that rule $\mathsf{i.CR2}$ implicitly covers the treatment of the top concept through its instances for which $K = \emptyset$. This point can best be clarified by an example. For instance, consider the normalised TBox $\mathcal{T} = \{A \sqsubseteq A, \top \sqsubseteq B\}$. We have $\mathcal{T} \models A \sqsubseteq B$, and thus completeness of the calculus for $\mathcal{EL}$ implies that the saturated TBox $\mathcal{T}^*$ must contain $A \sqsubseteq B$. To derive this GCI, the rule $\mathsf{CR2}$ is needed. In fact, $\mathsf{CR2}$ yields $A \sqsubseteq \top$, and then rule $\mathsf{CR3}$ can be used to obtain $A \sqsubseteq B$. In the calculus for $\mathcal{ELI}$, we start with the i.normalised TBox $\{\{A\} \sqsubseteq \{A\}, \emptyset \sqsubseteq \{B\}\}$. If we instantiate M with $\{A\}$, K with $\emptyset$, and C with $\{B\}$, then rule $\mathsf{i.CR2}$ yields $\{A\} \sqsubseteq \{B\}$ as required.

The rules $\mathsf{CR3}$ and $\mathsf{CR4}$ are obviously special cases of rule $\mathsf{i.CR2}$.

If one takes into account that $M_1 \sqsubseteq \forall r^-.\{A\}$ is equivalent to $\exists r.M_1 \sqsubseteq \{A\}$, the rule $\mathsf{i.CR3}$ looks similar to rule $\mathsf{CR5}$. Rule $\mathsf{i.CR3}$ realises transitivity through an existential restriction occurring on the right-hand side of one GCI and on the left-hand side of another GCI. One may wonder why, in the calculus for $\mathcal{EL}$, we need the rule $\mathsf{CR5}$ rather than the more restricted transitivity rule

$$\mathsf{CR5'} \quad \frac{A \sqsubseteq \exists r.A_1 \quad \exists r.A_1 \sqsubseteq B}{A \sqsubseteq B};$$

or, put the other way round, why the more restricted transitivity rule $\mathsf{i.CR3}$ is sufficient in the calculus for $\mathcal{ELI}$. Again, this is best explained by a simple example. For instance, consider the normalised TBox $\mathcal{T} = \{A \sqsubseteq \exists r.A_1, A_1 \sqsubseteq B_1, \exists r.B_1 \sqsubseteq B\}$. If we replace $\mathsf{CR5}$ in the calculus for $\mathcal{EL}$ by $\mathsf{CR5'}$, then $A \sqsubseteq B$ can no longer be derived. In the calculus for

 https://doi.org/10.1017/9781139025355.006

$\mathcal{ELI}$, we start with the i.normalised TBox

$$\{\{A\} \sqsubseteq \exists r.\{A_1\}, \{A_1\} \sqsubseteq \{B_1\}, \{B_1\} \sqsubseteq \forall r^-.\{B\}\}.$$

Applying rule i.CR2 to the second and the third GCI yields $\{A_1\} \sqsubseteq \forall r^-.\{B\}$. Now the rule i.CR3 can be applied to the first GCI in the above TBox and this derived GCI to obtain the desired GCI $\{A\} \sqsubseteq \{B\}$.

The rule i.CR4 does not have a corresponding rule in the calculus for $\mathcal{EL}$. It is required to deal with the additional expressive power caused by inverse roles, i.e, the fact that value restrictions on the right-hand side of GCIs can be expressed. Note that this is the only rule that can generate new sets of concept names other than singleton sets within $\mathcal{T}$-i.sequents: in fact, the set $M_2 \cup \{A\}$ may not have occurred in $\mathcal{T}'$ before.

This also shows that the algorithm for $\mathcal{ELI}$ runs in polynomial time if it receives a general $\mathcal{EL}$ TBox as input. Indeed, if we start with an $\mathcal{EL}$ TBox $\mathcal{T}_0$, then the corresponding i.normalised TBox $\mathcal{T}$ (written as a set of $\mathcal{T}$-i.sequents) contains only $\mathcal{T}$-i.sequents satisfying the following restrictions:

(i) the only sets occurring in these $\mathcal{T}$-i.sequents are the empty set and singleton sets;
(ii) value restrictions in these $\mathcal{T}$-i.sequents are only with respect to inverses of role names;
(iii) existential restrictions in these $\mathcal{T}$-i.sequents are only with respect to role names.

Let us call a $\mathcal{T}$-i.sequent satisfying these three restrictions an *$\mathcal{EL}$-$\mathcal{T}$-i.sequent.*

Lemma 6.27. *There are only polynomially many $\mathcal{EL}$-$\mathcal{T}$-i.sequents in the size of $\mathcal{T}$. In addition, applying an inference rule of Figure 6.3 to a set $\mathcal{T}'$ of $\mathcal{EL}$-$\mathcal{T}$-i.sequents yields a set of $\mathcal{EL}$-$\mathcal{T}$-i.sequents.*

Proof. The first statement of the lemma is obviously true since there are only polynomially many sets of concept names in $sig(\mathcal{T})$ of cardinality ≤ 1.

The only rule that could generate a $\mathcal{T}$-i.sequent violating the above three conditions is rule i.CR4. However, this rule is not applicable since it requires the same role name r or inverse of a role name $r = s^-$ to occur in both an existential restriction and a value restriction in $\mathcal{T}'$, which is prevented by the second and third conditions above. □

As an obvious consequence of this lemma, i.saturation terminates after a polynomial number of rule applications if applied to an i.normalised TBox that contains only $\mathcal{EL}$-$\mathcal{T}$-i.sequents.

Proposition 6.28. *The subsumption algorithm for $\mathcal{ELI}$ yields a polynomial-time decision procedure for subsumption in $\mathcal{EL}$.*

If we start with an $\mathcal{ELI}$ TBox whose i.normalisation does not yield a set of $\mathcal{EL}$-$\mathcal{T}$-i.sequents, then rule i.CR4 may cause the generation of an exponential number of $\mathcal{T}$-i.sequents, as illustrated by Example 6.19 above. However, though in this example the i.saturated TBox $\mathcal{T}^*$ indeed contains exponentially many $\mathcal{T}$-i.sequents, only a linear number of these $\mathcal{T}$-i.sequents is needed to derive the desired consequence $\{A\} \sqsubseteq \{B\}$. In the following example, one needs to derive exponentially many $\mathcal{T}$-i.sequents before the consequence $\{A\} \sqsubseteq \{B\}$ can be derived.

Example 6.29. Let A, B and $X_i, \overline{X}_i$ for $i = 0, \ldots, n-1$ be concept names and r a role name. Assume that $\mathcal{T}'$ consists of the following set of $\mathcal{T}$-i.sequents:

$$\begin{array}{rcl}
\{A\} & \sqsubseteq & \{\overline{X}_i\} \text{ for } 0 \leq i \leq n-1, \\
\emptyset & \sqsubseteq & \exists r.\emptyset, \\
\{\overline{X}_i, X_0, \ldots, X_{i-1}\} & \sqsubseteq & \forall r.\{X_i\} \text{ for } 0 \leq i \leq n-1, \\
\{X_i, X_0, \ldots, X_{i-1}\} & \sqsubseteq & \forall r.\{\overline{X}_i\} \text{ for } 0 \leq i \leq n-1, \\
\{\overline{X}_i, \overline{X}_j\} & \sqsubseteq & \forall r.\{\overline{X}_i\} \text{ for } 0 \leq j < i \leq n-1, \\
\{X_i, \overline{X}_j\} & \sqsubseteq & \forall r.\{X_i\} \text{ for } 0 \leq j < i \leq n-1, \\
\{X_0, \ldots, X_{n-1}\} & \sqsubseteq & \{B\}, \\
\{B\} & \sqsubseteq & \forall r^-.\{B\}.
\end{array}$$

Subsets of $\{X_i, \overline{X}_i \mid i = 0, \ldots, n-1\}$ containing exactly one of the concept names $X_i, \overline{X}_i$ for each $i, 0 \leq i < n$, can obviously be used to represent natural numbers k between 0 and $2^n - 1$. The set corresponding to the number k will be denoted as $X(k)$, i.e.,

$$\begin{array}{rcl}
X(0) & = & \{\overline{X}_0, \overline{X}_1, \ldots, \overline{X}_{n-1}\}, \\
X(1) & = & \{X_0, \overline{X}_1 \ldots, \overline{X}_{n-1}\}, \\
& \vdots & \\
X(2^n - 2) & = & \{\overline{X}_0, X_1, \ldots, X_{n-1}\}, \\
X(2^n - 1) & = & \{X_0, X_1, \ldots, X_{n-1}\}.
\end{array}$$

Using rule i.CR2 we can derive

$$\{A\} \sqsubseteq \exists r.\emptyset \quad \text{and} \quad \{A\} \sqsubseteq \forall r.\{X_0\}$$

as well as

$$\{A\} \sqsubseteq \forall r.\{\overline{X}_i\} \text{ for all } i, 0 < i < n.$$

Using n applications of i.CR4 we can thus derive

$$\{A\} \sqsubseteq \exists r.X(1).$$

Since $X(1)$ occurs in the TBox generated in this way, we can now use i.CR1 to derive

$$X(1) \sqsubseteq X_0 \quad \text{and} \quad X(1) \sqsubseteq \overline{X}_i \text{ for } i = 1, \ldots, n-1.$$

Thus, by applying the approach used above for $\{A\}$ to this set, we can derive $X(1) \sqsubseteq \exists r.X(2)$. Continuing this way, we obtain all the $\mathcal{T}$-i.sequents

$$X(k) \sqsubseteq \exists r.X(k+1) \text{ for } 1 \leq k \leq 2^n - 2.$$

Using the rule i.CR2, we can now derive

$$X(2^n - 1) \sqsubseteq \forall r^-.\{B\},$$

which together with $X(2^n - 2) \sqsubseteq \exists r.X(2^n - 1)$ yields $X(2^n - 2) \sqsubseteq \{B\}$ by an application of rule i.CR3. Continuing in this way, we can thus derive $X(1) \sqsubseteq \{B\}$, which then yields $X(1) \sqsubseteq \forall r^-.\{B\}$. Together with $\{A\} \sqsubseteq \exists r.X(1)$, we thus obtain

$$\{A\} \sqsubseteq \{B\}$$

by an application of rule i.CR3.

The derivation of $\{A\} \sqsubseteq \{B\}$ constructed above obviously has a length that is exponential in n, whereas the size of $\mathcal{T}'$ is polynomial in n. It is easy to see that there cannot be a derivation of this sequent that has polynomial length. In fact, one first needs to generate the exponentially many sequents $X(k) \sqsubseteq \exists r.X(k+1)$ for $1 \leq k \leq 2^n - 2$ before reaching B, which then has to be propagated back by generating the exponentially many sequents $X(k) \sqsubseteq \{B\}$ for $1 \leq k \leq 2^n - 1$.

6.3.2 A more abstract point of view

Both algorithms use inference rules to generate new GCIs that are consequences of the ones already obtained. This generation process is deterministic in the sense that GCIs, once added, are never removed. The two algorithms also have in common that it is sufficient to compute only consequences belonging to a certain *finite* set of *relevant* potential consequences, which is determined by the input TBox. Once all relevant consequences are computed, the subsumption query can be answered by a simple inspection of this set. The difference in the complexity of the two procedures stems from the fact that, for $\mathcal{EL}$, the cardinality of the set of relevant potential consequences is polynomial in the size of the input TBox, whereas it is exponential for $\mathcal{ELI}$.

From a semantic point of view, both algorithms generate *canonical models*, i.e., models $\mathcal{I}_{\mathcal{T}^*}$ of the normalised input TBox $\mathcal{T}$ in which subsumptions between concept names hold if and only if they follow from $\mathcal{T}$ (modulo certain occurrence restrictions formulated in the completeness results). For $\mathcal{EL}$, the domain of the canonical model consists of all the concept names occurring in the saturated TBox $\mathcal{T}^*$, the interpretation of the concept names is determined by the $\mathcal{T}$-sequents of the form $B \sqsubseteq A$ in $\mathcal{T}^*$, and the interpretation of the role names is determined by the $\mathcal{T}$-sequents of the form $A \sqsubseteq \exists r.B$ in $\mathcal{T}^*$. Similarly, for $\mathcal{ELI}$, the domain of the canonical model consists of all the sets of concept names occurring in the i.saturated TBox $\mathcal{T}^*$, the interpretation of the concept names is determined by the GCIs of the form $M \sqsubseteq \{A\}$ in $\mathcal{T}^*$, and the interpretation of the role names is determined by the GCIs of the form $M \sqsubseteq \exists r.N$ in $\mathcal{T}^*$.

In contrast to the type elimination algorithm for satisfiability in $\mathcal{ALC}$ with respect to general TBoxes, introduced in Chapter 5, the generation of the canonical model is a bottom-up procedure, i.e., it adds elements to the domain and to the extension of concepts and roles, rather than starting with a maximal set and successively removing elements.[2]

The tableau algorithms introduced in Chapter 4 compute a model of the TBox that refutes the subsumption in case it does not hold. But if the subsumption holds, then no model is computed. Another difference to the algorithms introduced in the present chapter is that the tableau algorithms are nondeterministic, i.e., different choices need to be made and backtracking is required if a decision was wrong.

The canonical model of an $\mathcal{ELI}$ TBox introduced in Definition 6.22 is not only a tool to show completeness of the classification algorithm for $\mathcal{ELI}$. It can also be employed to show other useful properties. As an example, we use the canonical model to show that $\mathcal{ELI}$ is convex. Intuitively, convexity says that $\mathcal{ELI}$ does not have any "hidden disjunctions":

[2] More formally speaking, type elimination computes a greatest fixpoint, whereas the algorithms introduced in the present chapter compute a least fixpoint.

Proposition 6.30. *$\mathcal{ELI}$ is* convex*, i.e., it satisfies the following* convexity property*: if $\mathcal{T}$ is an $\mathcal{ELI}$ TBox and $C, D_1, \ldots, D_n$ are $\mathcal{ELI}$ concepts, then*

$$\mathcal{T} \models C \sqsubseteq D_1 \sqcup \cdots \sqcup D_n \text{ if and only if } \mathcal{T} \models C \sqsubseteq D_i \text{ for some } i \in \{1, \ldots, n\}.$$

Note that the above definition of convexity makes sense even though $\mathcal{ELI}$ does not include disjunction as a constructor; in fact, the left-hand side of the above equivalence can simply be understood as a statement formulated in $\mathcal{ALCI}$.

Obviously, the DL $\mathcal{ALC}$ is not convex in the above sense as, for example, the TBox $\mathcal{T} = \{A \sqsubseteq B_1 \sqcup B_2\}$ satisfies $\mathcal{T} \models A \sqsubseteq B_1 \sqcup B_2$, but not $\mathcal{T} \models A \sqsubseteq B_i$ for any $i \in \{1, 2\}$. This is of course no surprise since $\mathcal{ALC}$ explicitly allows for disjunction.

However, things are not always that obvious. To see this, consider the DL $\mathcal{FLE}$, the extension of $\mathcal{EL}$ with value restrictions. In contrast to $\mathcal{ELI}$, an $\mathcal{FLE}$ TBox may have value restrictions on both the left- and the right-hand sides of GCIs. Despite not including disjunction as a concept constructor, this DL is not convex. To see this, take the TBox

$$\mathcal{T} = \{\exists r.\top \sqsubseteq B_1, \forall r.A \sqsubseteq B_2\}.$$

Then we have $\mathcal{T} \models \top \sqsubseteq B_1 \sqcup B_2$, but not $\mathcal{T} \models \top \sqsubseteq B_i$ for any $i \in \{1, 2\}$. In fact, the latter is easy to verify by giving a countermodel against the two subsumptions in question. To see the former, let $\mathcal{I}$ be a model of $\mathcal{T}$ and $d \in \Delta^{\mathcal{I}}$. Then either there is some $e \in \Delta^{\mathcal{I}}$ with $(d, e) \in r^{\mathcal{I}}$ or this is not the case. In the first case, $d \in (\exists r.\top)^{\mathcal{I}}$, thus $d \in B_1^{\mathcal{I}}$; in the second case, $d \in (\forall r.A)^{\mathcal{I}}$, thus $d \in B_2^{\mathcal{I}}$.

Convexity is of interest because reasoning algorithms for non-convex DLs typically need to employ nondeterminism or backtracking (such as tableau algorithms), or are best-case exponential (such as type elimination algorithms). They cannot be treated using consequence-based algorithms that are as simple and elegant as the ones presented in this chapter.

To prove Proposition 6.30, we first show a lemma, which will also turn out to be helpful in the next chapter.

Lemma 6.31. *Let $\mathcal{T}$ be an $\mathcal{ELI}$ TBox, C an $\mathcal{ELI}$ concept and Γ a finite set of $\mathcal{ELI}$ concepts. Then there is a model $\mathcal{I}$ of $\mathcal{T}$ and an element $d \in \Delta^{\mathcal{I}}$ such that the following holds for all concepts $D \in \Gamma$:*

$$\mathcal{T} \models C \sqsubseteq D \quad \text{if and only if} \quad d \in D^{\mathcal{I}}.$$

Proof. Let $\Gamma = \{D_1, \ldots, D_n\}$. We introduce new concept names $A, B_1, \ldots, B_n$ and extend $\mathcal{T}$ by GCIs that say that A is equivalent to C and B_i is equivalent to D_i $(i = 1, \ldots, n)$, i.e., we define

$$\mathcal{T}' := \mathcal{T} \cup \{A \sqsubseteq C, C \sqsubseteq A\} \cup \{B_i \sqsubseteq D, D \sqsubseteq B_i \mid i = 1, \ldots, n\}.$$

Let $\mathcal{S}$ be the i.normalised $\mathcal{ELI}$ TBox obtained from $\mathcal{T}'$ by applying the $\mathcal{ELI}$ normalisation rules, and let $\mathcal{S}^*$ be the i.saturated TBox obtained from $\mathcal{S}$ by an exhaustive application of the inference rules of Figure 6.3.

We define $\mathcal{I} := \mathcal{I}_{\mathcal{S}^*}$, i.e., $\mathcal{I}$ is the canonical interpretation induced by $\mathcal{S}^*$. By Lemma 6.24, $\mathcal{I}$ is a model of $\mathcal{S}^*$, and thus also of $\mathcal{S}$, $\mathcal{T}'$ and $\mathcal{T}$. Since A is a concept name occurring in $\mathcal{S}$, and $\mathcal{S}$ contains a GCI with left-hand side A, it is easy to see that $\{A\}$ occurs in $\mathcal{S}^*$. For this reason, $\{A\}$ belongs to $\Delta^\mathcal{I}$ and we can define $d := \{A\}$.

By Proposition 6.20, we have, for all $i = 1, \ldots, n$,

$$\mathcal{S} \models A \sqsubseteq B_i \ \text{ if and only if } \ \{A\} \sqsubseteq \{B_i\} \in \mathcal{S}^*,$$

and the definition of the canonical interpretation yields

$$\{A\} \sqsubseteq \{B_i\} \in \mathcal{S}^* \ \text{ if and only if } \ d = \{A\} \in B_i^\mathcal{I}.$$

Finally, the definition of $\mathcal{T}'$ and the fact that $\mathcal{S}$ is a conservative extension of $\mathcal{T}'$ yield

$$\mathcal{S} \models A \sqsubseteq B_i \ \text{ if and only if } \ \mathcal{T} \models C \sqsubseteq D_i.$$

To complete the proof, we observe that $B_i^\mathcal{I} = D_i^\mathcal{I}$ since $\mathcal{I}$ is known to be a model of $\mathcal{T}'$. □

Proof of Proposition 6.30. It is easy to see that Lemma 6.31 implies this proposition. In fact, the "if" direction of the definition of convexity is trivially satisfied. Thus consider the contrapositive of the "only if" direction, and assume that $\mathcal{T} \not\models C \sqsubseteq D_i$ for all $i \in \{1, \ldots, n\}$. Let $\Gamma = \{C, D_1, \ldots, D_n\}$. Then Lemma 6.31 yields a model $\mathcal{I}$ of $\mathcal{T}$ and a $d \in \Delta^\mathcal{I}$ such that $d \in C^\mathcal{I}$ and $d \notin D_i^\mathcal{I}$ for all $i \in \{1, \ldots, n\}$. Consequently, $\mathcal{T} \not\models C \sqsubseteq D_1 \sqcup \cdots \sqcup D_n$. □

6.4 Historical context and literature review

In the early times of DL research, people concentrated on identifying formalisms for which reasoning is tractable, i.e., can be performed in polynomial time. In addition, the presence of both conjunction and value restriction was seen as indispensable in a true DL. The DL with only these two concept constructors is called $\mathcal{FL}_0$ [Baa90]. It came as surprise to the community when Bernhard Nebel [Neb90b] was able to show that subsumption in $\mathcal{FL}_0$ is intractable (more precisely, coNP-complete) with respect to acyclic TBoxes. Actually, the complexity of the subsumption problem increases even further if the TBox formalism is extended: it is PSpace-complete with respect to cyclic TBoxes [Baa90, Baa96, KdN03] and even ExpTime-complete with respect to general TBoxes [BBL05]. These negative complexity results, together with the advent of practically efficient, though worst-case intractable, tableau-based algorithms, were the main reasons why the DL community for more than a decade basically abandoned the search for DLs with tractable inference problems, and concentrated on the design of practical tableau-based algorithms for expressive DLs.

The DL $\mathcal{EL}$ was first introduced in [BKM99] in the context of non-standard inferences in DLs. There, it was shown that subsumption between $\mathcal{EL}$ concepts (without a TBox) is polynomial. Several years later, this polynomiality result was first extended to subsumption with respect to acyclic and cyclic TBoxes [Baa03] and then to subsumption with respect to general TBoxes [Bra04]. The subsumption algorithm introduced in [Bra04] is quite similar to the one described in Section 6.1 above, though the basic data structures used to present it look different. The proof-theoretic subsumption algorithm in [Hof05] uses a presentation that is quite similar to the one employed in Section 6.1.

In addition to providing new theoretical insights into the complexity of reasoning in DLs, these algorithms also turned out to be relevant in practice. In fact, quite a number of biomedical ontologies are built using $\mathcal{EL}$. Perhaps the most prominent example is the well-known medical ontology SNOMED CT,[3] which comprises about 380,000 concepts and is used to generate a standardised healthcare terminology employed as a standard for medical data exchange in a variety of countries including the US, UK, Canada and Australia.

Interestingly, the polynomiality result for subsumption in $\mathcal{EL}$ with respect to general TBoxes is stable under the addition of several interesting means of expressivity, such as the bottom concept, nominals and role hierarchies [BBL05, BBL08]. The papers [BBL05, BBL08] show that adding certain other constructors to $\mathcal{EL}$ makes subsumption with respect to general TBoxes intractable or even undecidable. In particular, it is shown in [BBL08] that, in $\mathcal{ELI}$, subsumption with respect to

[3] `http://www.ihtsdo.org/snomed-ct/`

general TBoxes is ExpTime-complete. Nevertheless, the ideas underlying the polynomial-time subsumption algorithm for $\mathcal{EL}$ can be extended to $\mathcal{ELI}$. This was independently shown by Kazakov [Kaz09] and Vu [Vu08], actually for extensions of $\mathcal{ELI}$ that can express the medical ontology Galen.[4] The subsumption algorithm presented in Section 6.2 is similar to the one introduced in [Kaz09].

Regarding implementation, the Cel reasoner [BLS06], which basically implements the classification procedure introduced in [BBL05], was the first DL reasoner able to classify SNOMED CT in less than 30 minutes. More recent implementations of algorithms based on these ideas have significantly improved on these runtimes [LB10, Kaz09, KKS14], bringing the classification time down to a few seconds. The CB reasoner [Kaz09] was the first DL reasoner able to classify the full version of Galen.

As explained above, an important feature of $\mathcal{EL}$ and $\mathcal{ELI}$ is their convexity, because this is what enables practically efficient reasoning based on consequence-based algorithms. There are other interesting and relevant DLs that are convex, in particular Horn-$\mathcal{SHIQ}$ and its variations. Horn-$\mathcal{SHIQ}$ originates from a translation of the description logic $\mathcal{SHIQ}$ into disjunctive Datalog and can be understood as a maximal fragment of $\mathcal{SHIQ}$ that is convex [HMS07]. Essentially, Horn-$\mathcal{SHIQ}$ extends $\mathcal{ELI}$ with functional roles, the $\bot$ concept, role hierarchies and at-least restrictions $(\geqslant n\, r.C)$ on the right-hand side of GCIs. In fact, the consequence-based algorithm by Kazakov mentioned above [Kaz09] is able to handle Horn-$\mathcal{SHIQ}$. The "Horn" in the name Horn-$\mathcal{SHIQ}$ refers to the fact that this DL can be viewed as a fragment of first-order Horn logic and, indeed, any such fragment must be convex.

[4] http://www.opengalen.org/