

Practical Epistemic Entailment Checking in *SROIQ*

Anees Mehdi and Sebastian Rudolph

Institute AIFB, Karlsruhe Institute of Technology, DE
{anees.mehdi,sebastian.rudolph}@kit.edu

Abstract. In this paper, we present a reasoner capable of epistemic inferences in *SROIQ* knowledge bases. We first identify some counter intuitive effects of imposing the traditional semantics in epistemic extensions of expressive description logics (DLs). Thus, we provide a revised *downward compatible* semantics with a more intuitive behavior in such cases. Based on the new semantics, we present a reduction method for epistemic queries to standard DL reasoning. This enables us to deploy state-of-the-art DL reasoners for such non-standard inferences. Additionally, we provide an implementation of our approach and present first evaluation results.

1 Introduction

In the early 1980s, Hector J. Levesque questioned the adequateness of the query language in knowledge formalisms [6]. He proposed the idea of embedding the epistemic operator \mathbf{K} into a query language, thereby achieving the capabilities of knowledge base introspection. A similar line of research was initiated by R. Reiter in the context of databases [9]. Due to the extended reasoning capabilities, epistemic extensions have also been investigated (cf. e.g. [3, 2, 4]) in the context of Description Logics (DLs, cf. [1]).

To see the usefulness of the \mathbf{K} operator for epistemic querying, consider the following example. Assume we want to query for “known white wines that are not known to be produced in a French region” which can be solved by performing instance retrieval w.r.t. the epistemic DL concept $\mathbf{K}WhiteWine \sqcap \neg \exists \mathbf{K}locatedIn.\{FrenchRegion\}$. This query will not only retrieve the wines that are explicitly excluded from being French wines but also those for which there is just no evidence that they are French (neither directly nor indirectly via deduction). For a knowledge base containing $\{WhiteWine(MountadamRiesling), locatedIn(MountadamRiesling, AustralianRegion)\}$ as a subset, the query would yield *MountadamRiesling* as a result, whereas the same query without epistemic operators would produce an empty result. Moreover, by adding additional information such as *MountadamRiesling*

being located in a French region, the answer to the epistemic query would also become empty, which illustrates that introducing the epistemic operator into a logic brings about non-monotonicity.

Another typical use case for epistemic querying is integrity constraint checking: testing whether the axiom

$$\mathbf{K} Wine \sqsubseteq \exists \mathbf{K} hasSugar.\{Dry\} \sqcup \exists \mathbf{K} hasSugar.\{OffDry\} \sqcup \exists \mathbf{K} hasSugar.\{Sweet\}$$

is entailed allows to check whether for every named individual in the knowledge base that is known to be a wine it is also known (i.e. it can be logically derived from the ontology) what degree of sugar it has. Note that this cannot be taken for granted even if $Wine \sqsubseteq \exists hasSugar.\{Dry\} \sqcup \exists hasSugar.\{OffDry\} \sqcup \exists hasSugar.\{Sweet\}$ is stated in (or can be derived from) the ontology.

These examples illustrate an obvious added value of epistemic extensions of description logics in practical applications. However former research – focused on extending tableaux algorithms for less expressive languages – has not paced up with the development of reasoners for very expressive DLs. In fact, as we will discuss in the course of this paper, some expressive features like nominal concepts require special care when combined with the idea of introspection by epistemic operators.

This paper investigates the epistemic extension of the very expressive DL *SR_QIQ* [5]. When applying a semantics along the lines of [4] to *SR_QIQ* we observe effects that clearly contradict natural requirements for epistemic reasoning (that we call backward compatibility). This directly leads to the question for an altered semantics that “behaves well” also for *SR_QIQ*. We introduce such a semantics and show that it complies with the proposed requirements. With the more adequate semantics at hand, we then turn to the question of efficient algorithms for the specific problem of answering queries to classical (i.e., **K**-free) *SR_QIQ* ontologies. We solve this problem by providing a sound and complete reduction from epistemic querying to standard DL reasoning; our approach reduces occurrences of the **K** operator to intermediate calls to a standard DL reasoner. Employing this technique, existing reasoners for non-epistemic DLs can be reused in a black-box fashion for the task of answering epistemic queries. Based on this algorithm, we implemented a reasoner capable of answering epistemic queries to *SR_QIQ* knowledge bases. For the complete proofs and technical details, we refer the interested reader to the accompanying technical report [7].

2 Epistemic DLs and the Classical Semantics

We consider \mathcal{SROIQ} as the underlying DL (for details see [5]). The extension of \mathcal{SROIQ} with the epistemic operator \mathbf{K} , denoted by \mathcal{SROIQK} , allows \mathbf{K} to appear in front of concept or role expressions. We call a \mathcal{SROIQK} -role an *epistemic role* if \mathbf{K} occurs in it. An epistemic role is *simple* if it is of the form $\mathbf{K}S$ where S is a simple \mathcal{SROIQ} -role.

Following the way epistemic semantics for DLs have been hitherto defined (see, e.g., [4]), the classical semantics of \mathcal{SROIQK} is given as *possible world semantics* in terms of *epistemic interpretations*. Thereby the following two central assumptions are made. (1) *Common Domain Assumption*: all interpretations are defined over a fixed countably infinite domain Δ . (2) *Rigid Term Assumption*: for all interpretations, the mapping from individuals to domain elements is fixed (it is just the identity function). Due to these assumptions, we can w.l.o.g. stipulate $\Delta := N_I \cup N$. Essentially, these assumptions are imposed in order to ensure that (sets of) domain elements can be referred to and dealt with uniformly in a cross-domain manner.

Next, we provide the definition of epistemic interpretations. The main difference to the non-epistemic case, is that we provide a “context” of relevant models which are inspected whenever the extension of an epistemic concept or role is to be determined.

Definition 1. An *epistemic interpretation* for \mathcal{SROIQK} is a pair $(\mathcal{I}, \mathcal{W})$ where \mathcal{I} is a \mathcal{SROIQ} -interpretation and \mathcal{W} is a set of \mathcal{SROIQ} -interpretations, where \mathcal{I} and all of \mathcal{W} have the same infinite domain Δ with $N_I \subset \Delta$. The interpretation function $\cdot^{\mathcal{I}\mathcal{W}}$ is then defined as follows:

$$\begin{aligned}
a^{\mathcal{I}\mathcal{W}} &= a && \text{for } a \in N_I \\
X^{\mathcal{I}\mathcal{W}} &= X^{\mathcal{I}} && \text{for } X \in N_C \cup N_R \cup \{\top, \perp\} \\
\{a_1, \dots, a_n\}^{\mathcal{I}\mathcal{W}} &= \{a_1, \dots, a_n\} \\
(\mathbf{K}C)^{\mathcal{I}\mathcal{W}} &= \bigcap_{\mathcal{J} \in \mathcal{W}} (C^{\mathcal{J}\mathcal{W}}) && (\mathbf{K}R)^{\mathcal{I}\mathcal{W}} = \bigcap_{\mathcal{J} \in \mathcal{W}} (R^{\mathcal{J}\mathcal{W}}) \\
(C \sqcap D)^{\mathcal{I}\mathcal{W}} &= C^{\mathcal{I}\mathcal{W}} \cap D^{\mathcal{I}\mathcal{W}} && (C \sqcup D)^{\mathcal{I}\mathcal{W}} = C^{\mathcal{I}\mathcal{W}} \cup D^{\mathcal{I}\mathcal{W}} \\
(\neg C)^{\mathcal{I}\mathcal{W}} &= \Delta \setminus C^{\mathcal{I}\mathcal{W}} \\
(\exists R.\text{Self})^{\mathcal{I}\mathcal{W}} &= \{x \mid (x, x) \in R^{\mathcal{I}\mathcal{W}}\} \\
(\exists R.C)^{\mathcal{I}\mathcal{W}} &= \{x \mid \exists y. (x, y) \in R^{\mathcal{I}\mathcal{W}} \wedge y \in C^{\mathcal{I}\mathcal{W}}\} \\
(\forall R.C)^{\mathcal{I}\mathcal{W}} &= \{x \mid \forall y. (x, y) \in R^{\mathcal{I}\mathcal{W}} \rightarrow y \in C^{\mathcal{I}\mathcal{W}}\} \\
(\leq n R.C)^{\mathcal{I}\mathcal{W}} &= \{x \mid \#\{y \in C^{\mathcal{I}\mathcal{W}} \mid (x, y) \in R^{\mathcal{I}\mathcal{W}}\} \leq n\} \\
(\geq n R.C)^{\mathcal{I}\mathcal{W}} &= \{x \mid \#\{y \in C^{\mathcal{I}\mathcal{W}} \mid (x, y) \in R^{\mathcal{I}\mathcal{W}}\} \geq n\}
\end{aligned}$$

where C and D are \mathcal{SROIQK} -concepts and R is a \mathcal{SROIQK} -role. \diamond

From the above, one can see that \mathbf{KC} is interpreted as the set of objects that are in the extension of C under every interpretation in \mathcal{W} . This also makes clear why the common domain and rigid term assumption have to be imposed; otherwise the respective extension intersections would be empty. Note that the rigid term assumption implies the unique name assumption (UNA), i.e., for any epistemic interpretation $\mathcal{I} \in \mathcal{W}$ and for any two distinct individual names a and b , we have that $a^{\mathcal{I}} \neq b^{\mathcal{I}}$.

The notions of knowledge base, TBox, RBox and Abox, their respective axioms, and their interpretations can be extended from SROIQ to SROIQK in the obvious way.

An *epistemic model* for a SROIQK -knowledge base Σ is a *maximal* non-empty set \mathcal{W} of SROIQ -interpretations such that $(\mathcal{I}, \mathcal{W})$ satisfies Σ for each $\mathcal{I} \in \mathcal{W}$. A SROIQK -knowledge base Σ is said to be *satisfiable* if it has an epistemic model. The knowledge base Σ (*epistemically*) *entails* an axiom α (written $\Sigma \models \alpha$), if for every epistemic model \mathcal{W} of Σ , we have that for each $\mathcal{I} \in \mathcal{W}$, the epistemic interpretation $(\mathcal{I}, \mathcal{W})$ satisfies α . By definition, every SROIQ -knowledge base is a SROIQK -knowledge base. Note that a given SROIQ -knowledge base Σ has up to isomorphism only one unique epistemic model which is the set of all models of Σ having infinite domain and satisfying the unique name assumption. We denote this model by $\mathcal{M}(\Sigma)$.

3 Problems with the Classical Semantics

Following the intuition that led to the introduction of the \mathbf{K} operator as an extension of \mathbf{K} -free standard DL reasoning, a rather intuitive basic requirement to an epistemic DL semantics is arguably the following.

Definition 2. For a given DL \mathcal{L} , an epistemic DL semantics represented by an entailment relation \models is called *\mathcal{L} -backward-compatible* if it coincides with the (non-epistemic) standard semantics (represented by \models) on non-epistemic axioms, i.e. for an \mathcal{L} knowledge base Σ and an \mathcal{L} axiom α both of which not containing \mathbf{K} , we have $\Sigma \models \alpha$ exactly if $\Sigma \models \alpha$. Moreover, \models is called *\mathcal{L} -UNA-backward-compatible*, if $\Sigma \models \alpha$ exactly if $\Sigma \models \alpha$ under the unique name assumption. \diamond

We can show that \models is $\mathit{SRIQ}\setminus U$ -UNA-backward-compatible, where $\mathit{SRIQ}\setminus U$ denotes the description logic SROIQ without nominal concepts and the universal role. The main ingredient for this is the insight that for any finite interpretation of a given $\mathit{SRIQ}\setminus U$ knowledge base, we can come up with an infinite interpretation such that both interpretations behave in exactly the same way in terms of satisfaction of axioms.

Lemma 3. *Let Σ be a $\mathcal{SRIQ}\setminus U$ knowledge base. For any interpretation \mathcal{I} there is an interpretation \mathcal{I}_ω with infinite domain such that $\mathcal{I} \models \Sigma$ if and only if $\mathcal{I}_\omega \models \Sigma$.*

As a consequence, the restriction to infinite models imposed by the common domain assumption turns out to be not essential in the case of $\mathcal{SRIQ}\setminus U$. However, this situation changes drastically once nominals or the universal role are involved. To see this, consider the axioms $\top \sqsubseteq \{a, b, c\}$ or $\top \sqsubseteq \leq 3U.\top$. Each of these axioms considered as a knowledge base Σ has only models with at most three elements. Consequently, in both cases we have that Σ is unsatisfiable w.r.t. the classical epistemic semantics and consequently by *ex falso quodlibet* we, e.g., obtain $\Sigma \models \top \sqsubseteq \perp$ whereas we clearly have $\Sigma \not\models \top \sqsubseteq \perp$ even under the UNA. So we conclude that \models is not UNA-backward-compatible for any description logic that features nominals or simultaneously number restrictions and the universal role; in particular, it is not \mathcal{SROIQ} -UNA-backward-compatible.

While the imposed UNA may be a deliberate decision, we believe that non- \mathcal{SROIQ} -UNA-backward-compatibility of classical epistemic entailment is not intended but rather a side effect of a semantics crafted for and probed against less expressive description logics; it contradicts the intuition behind the **K** operator. This motivates our quest for a more appropriate, “domain-flexible” epistemic semantics. In [8], another approach, based on first-order logic (FOL), has been presented which circumvents the described problem by treating the equality as a congruence relation with minimized extension. However, the solution we present is closer to the original DL setting as it extends the standard definition of DL interpretations.

4 A Revisited Semantics

In order to allow for the necessary flexibility, we need to relinquish the common domain assumption and the rigid term assumption in the epistemic semantics: The domains we consider in the possible worlds should be allowed to have arbitrary (yet non-empty) size and be composed of arbitrary elements. An individual name may refer to different elements in different possible worlds. Also, individuals denoted by different individual names may coincide in some worlds but not in others.

Still, due to the reasons discussed before, we have to find a substitute for the common domain and rigid term assumptions as otherwise, every epistemic role or concept would have the empty set as extension.

We solve the problem by introducing one abstraction layer that assigns *abstract individual names* to domain elements. These abstract individual names are elements from $N_I \cup \mathbb{N}$ and hence common to all interpretations, thus they can serve as the “common ground” for different interpretations with different domains. We require that every domain element is associated with at least one abstract name, however, we also allow for different names denoting the same domain element (thus allowing for the possibility of finite domains). This intuition leads to the definition of extended interpretations.

Thereby, an *extended SROIQ-interpretation* \mathfrak{I} is a tuple $(\Delta_{\mathfrak{I}}, \cdot^{\mathfrak{I}}, \varphi_{\mathfrak{I}})$ such that

- $(\Delta_{\mathfrak{I}}, \cdot^{\mathfrak{I}})$ is a standard DL interpretation,
- $\varphi_{\mathfrak{I}} : N_I \cup \mathbb{N} \rightarrow \Delta^{\mathfrak{I}}$ is a surjective function from $N_I \cup \mathbb{N}$ onto $\Delta^{\mathfrak{I}}$, such that for all $a \in N_I$ we have that $\varphi_{\mathfrak{I}}(a) = a^{\mathfrak{I}}$.

Note that the function $\varphi_{\mathfrak{I}}$ returns the actual interpretation of an individual, given its (abstract) name, under the interpretation \mathfrak{I} . We lift $\varphi_{\mathfrak{I}}$ to sets of names and let $\varphi_{\mathfrak{I}}^{-1}$ denote the corresponding inverse. Based on the notion of extended interpretations, we define an *extended epistemic interpretation* for *SROIQK* as a pair $(\mathfrak{I}, \mathfrak{W})$, where \mathfrak{I} is an extended *SROIQ*-interpretation and \mathfrak{W} is a set of extended *SROIQ*-interpretations. Similar to epistemic interpretations, we define an extended interpretation function $\cdot^{\mathfrak{I}, \mathfrak{W}}$ as $\cdot^{\mathcal{I}, \mathcal{W}}$ in Definition 1 with the following modifications:

$$\begin{aligned} (\mathbf{KC})^{\mathfrak{I}, \mathfrak{W}} &= \varphi_{\mathfrak{I}} \left(\bigcap_{\mathfrak{J} \in \mathfrak{W}} \varphi_{\mathfrak{J}}^{-1} (C^{\mathfrak{I}, \mathfrak{W}}) \right) \\ (\mathbf{KR})^{\mathfrak{I}, \mathfrak{W}} &= \varphi_{\mathfrak{I}} \left(\bigcap_{\mathfrak{J} \in \mathfrak{W}} \varphi_{\mathfrak{J}}^{-1} (R^{\mathfrak{I}, \mathfrak{W}}) \right) \end{aligned}$$

Again, we set $[(\mathbf{KR})^{-}]^{\mathfrak{I}, \mathfrak{W}} := (\mathbf{KR}^{-})^{\mathfrak{I}, \mathfrak{W}}$ for an epistemic role $(\mathbf{KR})^{-}$.

The semantics of TBox, RBox and ABox axioms follows exactly that for the classical semantics. Here, instead of \models , we use the symbol \models_e , where e indicates that the relation is w.r.t. the extended semantics.

Like in case of the traditional (epistemic) semantics, we can define the notions of *extended epistemic modelhood* and the *satisfiability* of a *SROIQK*-knowledge base by considering extended interpretations instead of the standard DL interpretations. Similarly, the *entailment* (under the new semantics) of an axiom from a knowledge base can be defined.

We now first note that the newly established semantics has the desired compatibility property.

Theorem 4. \models_e is *SROIQ-backward-compatible*.

Proof sketch: First note that every satisfiable \mathbf{K} -free knowledge base Σ has exactly one extended epistemic model

$$\mathfrak{M}(\Sigma) = \{(\Delta_{\mathfrak{J}}, \cdot^{\mathfrak{J}}, \varphi_{\mathfrak{J}}) \mid (\Delta_{\mathfrak{J}}, \cdot^{\mathfrak{J}}) \models \Sigma, \varphi_{\mathfrak{J}} = \cdot^{\mathfrak{J}}|_{N_I} \cup f, f: \mathbb{N} \rightarrow \Delta_{\mathfrak{J}}\}.$$

Hence we have $\Sigma \models_{\mathfrak{e}} \alpha$ exactly if every $\mathfrak{J} \in \mathcal{M}_e(\Sigma)$ satisfies α , which (presuming α being \mathbf{K} -free) is the case exactly if $\Sigma \models \alpha$. \square

Consequently, this new semantics is more adequate for very expressive DLs such as \mathcal{SROIQ} . Yet, as will be shown later in the paper, it is also generic in the sense that for $\mathcal{SRIQ} \setminus U$ knowledge bases it behaves similar to the (classical) epistemic interpretation introduced earlier. With this new semantics, we avoid the aforementioned problems arising from nominals and the universal role in the language of a knowledge base. Arguably, this makes the revisited semantics a more suitable and appropriate choice for \mathbf{K} -extensions of expressive description logics, like \mathcal{SROIQK} .

5 Reducing Epistemic Querying to Standard DL Reasoning

We next introduce a novel technique for answering epistemic queries to \mathcal{SROIQ} knowledge bases under the revised semantics. More precisely, we provide a way of checking whether a given knowledge base entails concept assertions, role assertions or concept subsumptions where the involved concepts and roles may contain \mathbf{K} . Our method reduces this problem to a number of iterative entailment checks for \mathbf{K} -free axioms. To justify the translation, we establish two lemmata (c.f., Lemma 25 and Lemma 27 in the technical report) that characterize possible instances of epistemic concepts and roles, respectively. The idea is that the extension of a concept that is preceded by \mathbf{K} can only contain named individuals unless it comprises the whole domain. For roles we get a more intricate case distinction, however, it boils down to characterizing the set of “(inverse) role neighbors” of a fixed individual as the whole domain or a set of named individuals. As an “exceptional case” to this, we might get the diagonal of $\Delta^{\mathfrak{J}} \times \Delta^{\mathfrak{J}}$ as additional instances of an epistemic role.

Based on these characteristics of epistemic concepts and roles, we present a translation of epistemic concept expressions into equivalent \mathbf{K} -free ones. Note that the translation itself requires to check entailment of (\mathbf{K} -free) axioms, hence it is not strictly syntactical and it depends on the underlying knowledge base.

Definition 5. (Translation Function $\llbracket \cdot \rrbracket_{\Sigma}$)

Let Σ be a *SRIOIQ*-knowledge base. For a *SRIOIQ* concept A and a *SRIOIQ* role R , let $\text{trg}_{\Sigma}^{A,R}$ denote the nominal concept $\{a_1, \dots, a_n\}$ containing all a_i for which $\Sigma \models A \sqsubseteq \exists R.\{a_i\}$ and let $\text{trg}_{\Sigma}^{A,R} = \perp$ if there are no such a_i . We recursively define the function $\llbracket \cdot \rrbracket_{\Sigma}$ mapping *SRIOIQK* concept expressions to *SRIOIQ* concept expressions:

$$\begin{aligned}
\llbracket C \rrbracket_{\Sigma} &= C && \text{if } C \text{ is from } N_I \cup \{\top, \perp\}, \text{ a nominal,} \\
&&& \text{or a } \mathbf{K}\text{-free self concept;} \\
\llbracket C_1 \sqcap C_2 \rrbracket_{\Sigma} &= \llbracket C_1 \rrbracket_{\Sigma} \sqcap \llbracket C_2 \rrbracket_{\Sigma} \\
\llbracket C_1 \sqcup C_2 \rrbracket_{\Sigma} &= \llbracket C_1 \rrbracket_{\Sigma} \sqcup \llbracket C_2 \rrbracket_{\Sigma} \\
\llbracket \neg C \rrbracket_{\Sigma} &= \neg \llbracket C \rrbracket_{\Sigma} \\
\llbracket \exists R.D \rrbracket_{\Sigma} &= \exists R.\llbracket D \rrbracket_{\Sigma} && \text{for } \exists \in \{\forall, \exists, \geq n, \leq n\}, R \text{ } \mathbf{K}\text{-free} \\
\llbracket \mathbf{K}D \rrbracket_{\Sigma} &= \begin{cases} \top & \text{if } \Sigma \models \llbracket D \rrbracket_{\Sigma} \equiv \top \\ \{a \in N_I \mid \Sigma \models \llbracket D \rrbracket_{\Sigma}(a)\} & \text{otherwise} \end{cases} \\
\llbracket \exists \mathbf{K}.S.\text{Self} \rrbracket_{\Sigma} &= \llbracket \mathbf{K}\exists S.\text{Self} \rrbracket_{\Sigma} \\
\llbracket \exists \mathbf{K}R.D \rrbracket_{\Sigma} &= \exists R.\llbracket D \rrbracket_{\Sigma} && \text{for } \exists \in \{\forall, \exists, \geq n, \leq n\} \text{ and } \Sigma \models R \equiv U \\
\llbracket \forall \mathbf{K}P.D \rrbracket_{\Sigma} &= \neg \llbracket \exists \mathbf{K}P.\neg D \rrbracket_{\Sigma} \\
\llbracket \exists \mathbf{K}P.D \rrbracket_{\Sigma} &= \exists P.(\text{trg}_{\Sigma}^{\top, P} \sqcap \llbracket D \rrbracket_{\Sigma}) \sqcup (\text{trg}_{\Sigma}^{\top, P^-} \sqcap \exists P.\llbracket D \rrbracket_{\Sigma}) \\
&\quad \sqcup \underbrace{\bigsqcup_{a \in N_I} (\{a\} \sqcap \exists P.(\text{trg}_{\Sigma}^{\{a\}, P} \sqcap \llbracket D \rrbracket_{\Sigma})) \sqcup \llbracket D \rrbracket_{\Sigma}}_{\text{only if } \Sigma \models \top \sqsubseteq \exists P.\text{Self}} \\
\llbracket \leq n \mathbf{K}P.D \rrbracket_{\Sigma} &= \neg \llbracket \geq (n+1) \mathbf{K}P.D \rrbracket_{\Sigma} \\
\llbracket \geq n \mathbf{K}P.D \rrbracket_{\Sigma} &= \geq n P.(\text{trg}_{\Sigma}^{\top, P} \sqcap \llbracket D \rrbracket_{\Sigma}) \sqcup (\text{trg}_{\Sigma}^{\top, P^-} \sqcap \geq n P.\llbracket D \rrbracket_{\Sigma}) \\
&\quad \sqcup \bigsqcup_{a \in N_I} (\{a\} \sqcap \geq n P.(\text{trg}_{\Sigma}^{\{a\}, P} \sqcap \llbracket D \rrbracket_{\Sigma})) \\
&\quad \sqcup \underbrace{(\neg \{a \mid a \in N_I\} \sqcap \llbracket D \rrbracket_{\Sigma} \sqcap \geq (n-1) P.(\text{trg}_{\Sigma}^{\top, P} \sqcap \llbracket D \rrbracket_{\Sigma}))}_{\text{only if } \Sigma \models \top \sqsubseteq \exists P.\text{Self}} \quad \diamond
\end{aligned}$$

Observe that by definition, the result of applying this function to an epistemic concept indeed yields a concept not containing \mathbf{K} . Moreover the following lemma, which can be proved by structural induction over the concept expression, ensures that the translation function preserves the concept extension.

Lemma 6. *Let Σ be a *SRIOIQ*-knowledge base and C be a *SRIOIQK* concept. Then for any extended interpretation $\mathfrak{I} \in \mathfrak{M}(\Sigma)$, we have that $C^{\mathfrak{I}, \mathfrak{M}(\Sigma)} = \llbracket C \rrbracket_{\Sigma}^{\mathfrak{I}, \mathfrak{M}(\Sigma)}$.*

Consequently this lemma can be employed to prove our main result justifying our approach of deciding entailment of epistemic axioms based on non-epistemic standard reasoning.

Theorem 7. *For a *SRIOIQ*-knowledge base Σ , *SRIOIQK* concept C , D , and an individual a , the following hold:*

1. $\Sigma \models_e C(a)$ if and only if $\Sigma \models \llbracket C \rrbracket_\Sigma(a)$.
2. $\Sigma \models_e C \sqsubseteq D$ if and only if $\Sigma \models \llbracket C \rrbracket_\Sigma \sqsubseteq \llbracket D \rrbracket_\Sigma$.

Finally, we are also able to establish the correspondence that the classical and the newly introduced semantics coincide, as far as epistemic querying on $SRIQ\setminus U$ knowledge bases is concerned. This result further substantiates our claim that our semantics is a natural extension of the original intuition behind epistemic DLs.

Theorem 8. *Let Σ be a $SRIQ\setminus U$ knowledge base, C and D $SR\text{OIQK}$ concepts, and a an individual name. Then, the following hold:*

1. $\Sigma \models_e C(a)$ under the unique name assumption if and only if $\Sigma \models C(a)$.
2. $\Sigma \models_e C \sqsubseteq D$ under the unique name assumption if and only if $\Sigma \models C \sqsubseteq D$.

This can be proved by providing a transformation function similar to $\llbracket \cdot \rrbracket_\Sigma$ for the classical semantics, proving its correctness and showing that it coincides with $\llbracket \cdot \rrbracket_\Sigma$ on $SRIQ\setminus U$ knowledge bases.

6 A System

Based on the results established in the preceding section, we have implemented a preliminary prototype. The system takes an epistemic concept as input and translates it into an equivalent non-epistemic one according to Definition 5. A detailed system description is provided in the technical report. A running system has been uploaded and shared on google-code¹. For the purpose of testing, we consider two versions of the wine ontology² with 483 and 1127 individuals. As a measure, we consider the translation time of an epistemic concept to a non-epistemic equivalent one and the instance retrieval time of the translated concept. We consider different epistemic concepts. For each such concept C , we consider a non-epistemic concept obtained from C by dropping the \mathbf{K} -operators from it (see Table 1). Given a concept C , $t_{(C)}$ and $|C_i|$ represent the time in seconds required to compute the instances and the number of instances computed for C_i . Finally for an epistemic concept EC_i , $t_{\top(EC_i)}$ represents the time required to translate EC_i to its non-epistemic equivalent. Table 2 provides our evaluation results. From Table 2, the time

¹ <http://code.google.com/p/epistemicdl/>

² <http://www.w3.org/TR/owl-guide/wine.rdf>

Table 1. Concepts used for instance retrieval experiments.

C_1	$\exists hasWineDescriptor. WineDescriptor$
EC_1	$\exists \mathbf{K} hasWineDescriptor. \mathbf{K} WineDescriptor$
C_2	$\forall hasWineDescriptor. WineDescriptor$
EC_2	$\forall \mathbf{K} hasWineDescriptor. \mathbf{K} WineDescriptor$
C_3	$\exists hasWineDescriptor. WineDescriptor \sqcap \exists madeFromFruit. WineGrape$
EC_3	$\exists \mathbf{K} hasWineDescriptor. \mathbf{K} WineDescriptor \sqcap \exists \mathbf{K} madeFromFruit. \mathbf{K} WineGrape$
C_4	$WhiteWine \sqcap \neg \exists locatedIn. \{ FrenchRegion \}$
EC_4	$\mathbf{K} WhiteWine \sqcap \neg \exists \mathbf{K} locatedIn. \{ FrenchRegion \}$
C_5	$Wine \sqcap \neg \exists hasSugar. \{ Dry \} \sqcap \neg \exists hasSugar. \{ OffDry \} \sqcap \neg \exists hasSugar. \{ Sweet \}$
EC_5	$\mathbf{K} Wine \sqcap \neg \exists \mathbf{K} hasSugar. \{ Dry \} \sqcap \neg \exists \mathbf{K} hasSugar. \{ OffDry \} \sqcap \neg \mathbf{K} \exists hasSugar. \{ Sweet \}$

Table 2. Evaluation

Ontology	Concept	$t_{(C_i)}$	$ C_i $	Concept	$t_{\mathbf{T}(EC_i)}$	$t_{(EC_i)}$	$ EC_i $
Wine 1	C_1	2.13	159	EC_1	46.98	0.04	3
	C_2	0.01	483	EC_2	0.18	0.00	0
	C_3	28.90	159	EC_3	79.43	6.52	3
	C_4	0.13	0	EC_4	95.60	107.82	72
	C_5	52.23	80	EC_5	60.78	330.49	119
Wine 2	C_1	8.51	371	EC_1	351.78	0.13	308
	C_2	0.30	1127	EC_2	0.127	0.00	0
	C_3	227.10	371	EC_3	641.24	19.58	7
	C_4	0.34	0	EC_4	865.04	840.97	168
	C_5	295.87	240	EC_5	381.41	2417.65	331

required to compute the number of instances is feasible; it is roughly in the same order of magnitude as for non-epistemic concepts. Note also that the runtime comparison between epistemic concepts EC_i and their non-epistemic counterparts C_i should be taken with a grain of salt as they are semantically different in general, as also indicated by the fact that there are cases where retrieval for the epistemic concept takes less time than for the non-epistemic version. As a general observation, we noticed that instances retrieval for an epistemic concept where a \mathbf{K} -operator occurs within the scope of a negation, tends to require much time.

7 Conclusion and Outlook

We argued how the traditional semantics for epistemic DLs causes problems and thus suggested a revision to the semantics. We proved that this revised semantics solves the aforementioned problem while coinciding with the traditional semantics on less expressive DLs (up to $\mathcal{SRIQ}\setminus U$). Focusing on the new semantics, we provided a way of answering epistemic

queries to *SRIOQ* knowledge bases via a reduction to a series of standard reasoning steps. Finally, we presented an implementation allowing for epistemic querying in *SRIOQ*.

Avenues for future research include the following: First, we will investigate to what extent the methods described here can be employed for entailment checks on *SRIOQK* knowledge bases, i.e., in cases where **K** occurs inside the knowledge base. In that case, stronger non-monotonic effects occur and the unique-epistemic-model property is generally lost. On the more practical side, we aim at further developing our initial prototype. We are confident that by applying appropriate optimizations such as caching strategies and syntactic query preprocessing a significant improvement in terms of runtime can be achieved. In the long run, we aim at demonstrating the added value of epistemic querying by providing an appropriate user-front-end and performing user studies. Furthermore, we will propose an extension of the current OWL standard by epistemic constructs in order to provide a common ground for future applications.

References

1. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press (2003)
2. Donini, F., Nardi, D., Rosati, R.: Non-first-order features in concept languages. In: Proceedings of the Fourth Conference of the Italian Association for Artificial Intelligence (AI*IA'95). pp. 91–102. Springer (1995)
3. Donini, F.M., Lenzerini, M., Nardi, D., Schaerf, A., Nutt, W.: Adding epistemic operators to concept languages. In: Bernhard Nebel, Charles Rich, W.R.S. (ed.) Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning (KR'92). pp. 342–353. Morgan Kaufmann (1992)
4. Donini, F.M., Lenzerini, M., Nardi, D., Schaerf, A., Nutt, W.: An epistemic operator for description logics. Artificial Intelligence 100(1-2), 225–274 (1998)
5. Horrocks, I., Kutz, O., Sattler, U.: The even more irresistible *SRIOQ*. In: Doherty, P., Mylopoulos, J., Welty, C.A. (eds.) Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning (KR'06). pp. 57–67. AAAI Press (2006)
6. Levesque, H.J.: Foundations of a functional approach to knowledge representation. Artificial Intelligence 23(2), 155–212 (1984)
7. Mehdi, A., Rudolph, S.: Revisiting semantics for epistemic extensions of description logics. Technical Report 3015, Institute AIFB, Karlsruhe Institute of Technology (2011), available at <http://www.aifb.kit.edu/web/Techreport3015/en>
8. Motik, B., Rosati, R.: Reconciling description logics and rules. J. ACM 57(5) (2010)
9. Reiter, R.: What should a database know? Journal of Logic Programming 14(1-2), 127–153 (1992)