On Improving SAT-Based Axiom Pinpointing of $\mathcal{EL}^+$ Ontologies

M. Fareed Arif[1] and Joao Marques-Silva[1,2]

[1] CASL, University College Dublin, Ireland
muhammad.arif.1@ucdconnect.ie, jpms@ucd.ie
[2] INESC-ID, IST, ULisboa, Portugal

Abstract. The $\mathcal{EL}$ family of Description Logics, and particularly $\mathcal{EL}^+$, is very relevant from a practical perspective, finding application in life science informatics, including the design of medical ontologies, with both $\mathcal{EL}$ and $\mathcal{EL}^+$ known to be tractable. Modeling an ontology is a difficult and error-prone process, which may introduce several unintended subsumption relations. Finding explanations for such subsumption relations is referred to as axiom pinpointing and constitutes a major step in debugging ontologies. Concretely, it consists in computing a set-wise minimal set of axioms that explains the reason for a subsumption relation. Recent work has devised an encoding of the classification of an $\mathcal{EL}^+$ ontology to a polynomial-size Horn propositional formula. This result enabled the development of a method for axiom pinpointing based on the analysis of the Horn formula, yielding clear performance gains over previous approaches. Building on this earlier work, but also on an large body of research in the analysis of unsatisfiable propositional formulas, this paper proposes a new algorithm that exploits an important relationship between minimal axiom sets and minimal unsatisfiable subformulas in the propositional domain. Experimental results show substantial performance improvements over previous methods in the literature.

1 Introduction

The lightweight Description Logics (DLs) $\mathcal{EL}$ and $\mathcal{EL}^+$ have attracted a considerable interest in the recent past [3,2,9,21,28,17], mainly due to the fact that they have efficient decision algorithms and, although exhibiting a limited expressiveness, they have been proven useful in important application domains, including the design of medical ontologies.

Building an ontology is a difficult task in which errors may be introduced. Consequently, undesired subsumption relations may be entailed by the resulting ontology. In this scenario, finding explanations for the subsumption relation can be very useful for debugging the ontology. This problem is termed *axiom pinpointing*. Concretely, it consists of computing subset-minimal axiom sets (MinAs) that explain the subsumption relation. Axiom pinpointing has been studied extensively over the last decade [32,27,7,33,8,9], and even some approaches can be traced to the mid 90s [4]. Regarding the $\mathcal{EL}$ family of DLs, it is worth mentioning the label-based classification algorithm proposed in [7,8] for axiom pinpointing. This approach generates a (worst-case exponential-size) propositional formula and computes all MinAs by finding its minimal models, which is an NP-hard problem. More recently, in [34,35] the authors devised a polynomial-size encoding of the classification of an $\mathcal{EL}^+$ ontology into a Horn propositional formula (i.e. it can be exponentially more compact than earlier work). In same work, this propositional encoding is exploited by an axiom pinpointing approach based on propositional satisfiability (SAT) solving [39,26] and SMT-like techniques [18]. Although effective at computing minimal axiom sets, these dedicated algorithms often fail to enumerate all MinAs to completion, or proving that no additional MinA exists. The computation of MinAs can be greatly improved by using optimizations, such as the cone-of-influence (COI) reduction technique [35], based on well-known results in model checking [14], which can simplify the encoded propositional formula to a great extent. This technique is related with $\mathcal{EL}^+$ reachability-based modularization [9].

Building on previous work, the main contribution of this paper is to show that the computation of MinAs can be related with the extraction of a minimal unsatisfiable subformula (MUS) of the Horn formula encoding [34,35]. The relationship between MUSes and MinAs makes it possible to benefit from the large recent body of work on extracting MUSes, but also minimal correction subsets (MCSes), as well as their minimal hitting set relationship [31,10,20], which for the propositional case allows for exploiting the performance of modern SAT solvers. An extensive experimental study shows that the new approach substantially outperforms the state-of-the-art methods in the literature, namely CEL [5,9] and EL2SAT [34,35].

The remainder of the paper is structured as follows. Section 2 presents the background and notation used throughout the paper. In Section 3.1, a brief description of the propositional Horn encoding of $\mathcal{EL}$ and $\mathcal{EL}^+$ ontologies is provided. The proposed new $\mathcal{EL}^+$ axiom pinpointing approach is introduced in Section 4. The experimental results are reported in Section 5. Finally, Section 6 concludes the paper and outlines some lines of future research.

2 Preliminaries

This section introduces the background and notation used throughout the paper, both related with Lightweight Description Logic (Section 2.1) and Boolean Satisfiability (SAT) (Section 2.2). Finally, we present the standard definitions and analysis of unsatisfiable formulas (Section 2.3).

2.1 Lightweight Description Logic $\mathcal{EL}^+$

The syntax and semantics of common logical constructs for the family $\mathcal{EL}$ family of DLs [3,2] are listed in Table 1. An ontology consists of a set of concept names $\mathsf{N_C}$ and a set of role names $\mathsf{N_R}$, which inductively define concept descriptions. Every concept name $C \in \mathsf{N_C}$ and the top concept $\top$ is a concept description. If C and D are concept descriptions then $C \sqcap D$ is also a concept description. If C is a concept description and $r \in \mathsf{N_R}$ then $\exists r.C$ is also a concept description. The set of concept descriptions is denoted as $\mathcal{CS}$. In an ontology, a general concept inclusion (GCI) is subclass relationship of form $C \sqsubseteq D$ where $C, D \in \mathcal{CS}$ and a role inclusion axiom[1] is of form

[1] A role inclusion can be a reflexive, transitive or an identity relation.

Syntax	Semantics
$\top$	$\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$
C	$C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ for each $C \in \mathsf{N_C}$
$C \sqcap D$	$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$
$\exists r.C$	$(\exists r.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}}, (x, y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
$r_1 \circ \cdots \circ r_n \sqsubseteq s$	$r_1^{\mathcal{I}} \circ \cdots \circ r_n^{\mathcal{I}} \subseteq s^{\mathcal{I}}$

Table 1: Syntax and Semantics of $\mathcal{EL}$ family of DL.

NR_1	$C \equiv D \rightsquigarrow C \sqsubseteq D, D \sqsubseteq C$ where $C, D \in \mathsf{N_C}^{\top}$
NR_2	$C \sqsubseteq D \sqcap F \rightsquigarrow C \sqsubseteq D, C \sqsubseteq F$ where $C, D, F \in \mathsf{N_C}^{\top}$
NR_3	$\hat{C} \sqcap D \sqsubseteq F \rightsquigarrow E \sqsubseteq \hat{C}, E \sqcap D \sqsubseteq F$ where $\hat{C} \in \mathcal{CS}, D, F \in \mathsf{N_C}^{\top} \cup \{E\}$
NR_4	$\exists r.\hat{C} \sqsubseteq D \rightsquigarrow \exists r.E \sqsubseteq D, E \sqsubseteq \hat{C}$ where $\hat{C} \in \mathcal{CS}, D \in \mathsf{N_C}^{\top} \cup \{E\}$
NR_5	$C \sqsubseteq \exists r.\hat{D} \rightsquigarrow C \sqsubseteq \exists r.E, E \sqsubseteq \hat{D}$ where $\hat{D} \in \mathcal{CS}, C \in \mathsf{N_C}^{\top} \cup \{E\}$
NR_6	$r_1 \circ \cdots \circ r_m \sqsubseteq s \rightsquigarrow r_1 \circ \cdots \circ r_{m-1} \sqsubseteq u, u \circ r_m \sqsubseteq s$ where $r_1, \ldots, r_m, u, s \in \mathsf{N_R}$

Table 2: Description logic $\mathcal{EL}^+$ Normalization Rules.

$r_1 \circ \cdots \circ r_n \sqsubseteq s$ where $r_1, \ldots, r_n, s \in \mathsf{N_R}$. An $\mathcal{EL}$ ontology only contains a finite set of GCIs. Additionally, $\mathcal{EL}^+$ ontology contains role inclusion axioms. An ontology $\mathcal{O}$ is called unfold-able if the left hand side of the axioms is a concept name and the right hand side, the definition, does not contain a reference to the defined concept. In description logic, an unfold-able ontology $\mathcal{O}$ can be normalized by applying the normalization rules shown in Table 2. The normalized ontology is of polynomial size and can be computed in linear time [13]. During normalization, the set of concept names $\mathsf{N_C}$ of any given ontology $\mathcal{O}$ is augmented with $\top$ concept, denoted as $\mathsf{N_C}^{\top}$, and freshly introduced concept names. Every normalized ontology $\mathcal{O}$ contains GCI's of following form:

$$
\begin{aligned}
C &\sqsubseteq D \\
C_1 \sqcap \cdots \sqcap C_n &\sqsubseteq D \\
C &\sqsubseteq \exists r.D \\
\exists r.C &\sqsubseteq D
\end{aligned}
$$

where $C, D \in \mathsf{N_C}$, $r \in \mathsf{N_R}$ and $n \geq 1$. An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \circ^{\mathcal{I}})$ consists of a non-empty domain set $\Delta^{\mathcal{I}}$ and a mapping function $\circ^{\mathcal{I}}$ that relates each concept $C \in \mathsf{N_C}$ to a set $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and each role name $r \in \mathsf{N_R}$ to relation $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. An interpretation $\mathcal{I}$ is a model of an ontology $\mathcal{O}$ if and only if it satisfies all the GCIs in $\mathcal{O}$ as described in Table 1. An ontology $\mathcal{O}$ is consistent if it has at least one model and a concept C is satisfiable w.r.t. $\mathcal{O}$ if there exists a model $\mathcal{I}$ of $\mathcal{O}$ such that $C^{\mathcal{I}} \neq \emptyset$. The subsumption problem $C \sqsubseteq_{\mathcal{O}} D$ w.r.t. an ontology $\mathcal{O}$ is an important inference problem considered in the $\mathcal{EL}$ family of DLs and defined as follows [8].

Definition 1 (Concept Subsumption). *For any given ontology $\mathcal{O}$ and concept descriptions $C, D \in \mathcal{CS}$, the subsumption relation $C \sqsubseteq_{\mathcal{O}} D$ holds (i.e., D subsumes C w.r.t. $\mathcal{O}$) if and only if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for every model of $\mathcal{O}$.*

A *classification* of a normalized ontology $\mathcal{O}$ represents all subsumption relations between concept names in $\mathcal{O}$. A subsumption relation is called an *assertion* and a set of assertions, denoted as $\mathcal{A}$, comprises of every existing and implied implied subsump-

	Normalized $\mathcal{O}$	Mapping Function (S and R)	$\mathcal{A} \rightsquigarrow \mathcal{A}'$
CR_0	$C \sqsubseteq D \in \mathcal{O}$	If $C \subseteq S(X), D \notin S(X)$ then $S(X) := S(X) \cup \{D\}$	$X \sqsubseteq C, C \sqsubseteq D \rightsquigarrow X \sqsubseteq D$
CR_1	$C_1 \sqcap C_2 \sqsubseteq D \in \mathcal{O}$	If $C_1, C_2 \subseteq S(X), D \notin S(X)$ then $S(X) := S(X) \cup \{D\}$	$X_1 \sqsubseteq C_1, X_2 \sqsubseteq C_2 \rightsquigarrow X \sqsubseteq D$
CR_2	$C \sqsubseteq \exists r.D \in \mathcal{O}$	If $C \in S(X), (X,D) \in R(r)$ then $R(r) := R(r) \cup \{(X,D)\}$	$X \sqsubseteq C \rightsquigarrow X \sqsubseteq \exists r.D$
CR_3	$\exists r.C \sqsubseteq D \in \mathcal{O}$	If $(X,Y) \in R(r), C \in S(Y), D \notin S(X)$ then $S(X) := S(X) \cup \{D\}$	$X \sqsubseteq \exists r.E, E \sqsubseteq C \rightsquigarrow X \sqsubseteq D$
CR_4	$r \sqsubseteq s \in \mathcal{O}$	If $(X,D) \in r$ and $(X,D) \notin R(s)$ then $R(s) := R(s) \cup \{(X,D)\}$	$X \sqsubseteq \exists r.D \rightsquigarrow X \sqsubseteq \exists s.D$
CR_5	$r \circ s \sqsubseteq t \in \mathcal{O}$	If $\{(X,E) \in R(r)$ and $(E,D)\} \in R(s)$ then $R(t) := R(t) \cup \{(X,D)\}$	$X \sqsubseteq \exists r.E_1, E_2 \sqsubseteq \exists s.D \rightsquigarrow X \sqsubseteq \exists t.D$

Table 3: Description logic $\mathcal{EL}^+$ Completion Rules.

tion relation of a normalized ontology $\mathcal{O}$. For $\mathcal{EL}$ family of DLs, the subsumption and classification problems can be solved in polynomial time [13].

For any given ontology $\mathcal{O}$, the $\mathcal{EL}^+$ subsumption algorithm [3] initializes the assertion set $\mathcal{A}$ by adding trivial GCI's $C \sqsubseteq C$, $C \sqsubseteq \top$ for every concept $C \in \mathsf{N_C}$ and extends $\mathcal{A}$ by applying completion rules described in Table 3. The completion rules are defined using the following concept mapping function $S : {\mathsf{N_C}}^{\top} \mapsto 2^{{\mathsf{N_C}}^{\top}}$ and the role mapping function $R : {\mathsf{N_R}}^{\top} \mapsto 2^{{\mathsf{N_C}}^{\top} \times {\mathsf{N_C}}^{\top}}$ where ${\mathsf{N_C}}^{\top}, {\mathsf{N_R}}^{\top}$ are set of concept and role names augmented by $\top$ as defined in [3,2]. The process of applying rules terminates in case no more rules can be applied (i.e., no additional assertions can be entailed in $\mathcal{A}$). For concepts $C, D \in {\mathsf{N_C}}^{\top}$, the subsumption relation $C \sqsubseteq_{\mathcal{O}} D$ holds if and only if $D \in S(C)$ (i.e., $C \sqsubseteq D \in \mathcal{A}$) and $C \sqsubseteq_{\mathcal{O}} \exists r.D$ holds if and only if $(C, D) \in R(r)$ (i.e., $C \sqsubseteq \exists r.D \in \mathcal{A}$).

Theorem 1 (Theorem 1 in [8]). *Given any normalized ontology $\mathcal{O}$ in $\mathcal{EL}^+$, the subsumption algorithm terminates in time polynomial in the size of $\mathcal{O}$ (i.e., $|\mathcal{A}| := |\,{\mathsf{N_C}}^{\top}|^2 \times |\,{\mathsf{N_R}}^{\top}|$). The final assertion set $\mathcal{A}$ satisfies $C \sqsubseteq_{\mathcal{O}} D$ iff $C \sqsubseteq D \in \mathcal{A}$ (i.e., $D \in S(C)$) for all concept names $C, D \in {\mathsf{N_C}}^{\top}$ in $\mathcal{O}$.*

2.2 Propositional Satisfiability

Standard propositional satisfiability (SAT) definitions are assumed [11]. This includes standard definitions for variables, literals, clauses and CNF formulas. Formulas are represented by $\mathcal{F}$, $\mathcal{M}$, $\mathcal{M}'$, $\mathcal{C}$ and $\mathcal{C}'$, but also by ϕ and ψ. Horn formulae are such that every clause contains at most one positive literal. It is well-known that SAT on Horn formulae can be decided in linear time [15]. The paper explores both MUSes and MCSes of propositional formulae.

Definition 2 (MUS). *$\mathcal{M} \subseteq \mathcal{F}$ is a* Minimal Unsatisfiable Subformula *(MUS) of $\mathcal{F}$ iff $\mathcal{M}$ is unsatisfiable and $\forall_{\mathcal{M}' \subsetneq \mathcal{M}}\, \mathcal{M}'$ is satisfiable.*

Definition 3 (MUS). *$\mathcal{C} \subseteq \mathcal{F}$ is a* Minimal Correction Subformula *(MCS) of $\mathcal{F}$ iff $\mathcal{F} \setminus \mathcal{C}$ is satisfiable and $\forall_{\mathcal{C}' \subsetneq \mathcal{C}}\, \mathcal{F} \setminus \mathcal{C}'$ is unsatisfiable.*

A well-known result, which will be used in the paper is the minimal hitting set relationship between MUSes and MCSes of an unsatisfiable formula $\mathcal{F}$ [31,12,10,20].

Theorem 2. *Let $\mathcal{F}$ be unsatisfiable. Then,*

1. *Each MCS of $\mathcal{F}$ is a minimal hitting set of the MUSes of $\mathcal{F}$.*
2. *Each MUS of $\mathcal{F}$ is a minimal hitting set of the MCSes of $\mathcal{F}$.*

Maximum Satisfiability (MaxSAT) is an important problem related to unsatisfiable formulas and consists of finding an assignment that maximizes the number of satisfied clauses. In *partial* MaxSAT, a formula φ is partitioned into a set of hard (φ_H) and soft (φ_S) clauses, i.e. $\varphi = \varphi_H \cup \varphi_S$. Hard clauses must be satisfied while soft clauses can be relaxed. The MaxSAT solution represents the smallest MCS in the formula. We have used partial MaxSAT encoding and MUS enumeration (Section 2.3) in our proposed axoim pinpointing method (Section 4).

2.3 Enumeration of MUSes

The problem of enumerating all the MUSes of an unsatisfiable formula has been studied in different settings [31,12,10,20,19,29], starting with the seminal work of Reiter [31]. Although the enumeration of MCSes can be achieved in a number of different ways [22], the enumeration of MUSes is believed to be far more challenging. Intuitively, the main difficulty is how to block one MUS and then use a SAT solver (or some other decision procedure) to compute the next MUS. This difficulty with MUS enumeration motivated a large body of work exploiting a fundamental relationship between MUSes and MCSes. Indeed, it is well-known (e.g. see Theorem 2) that MCSes are minimal hitting sets of MUSes, and MUSes are minimal hitting sets of MCSes [31,12,10,20]. As a result, early approaches [10,20] for enumeration of MUSes were organized as follows: (i) enumerate all MCSes of a formula; (ii) compute the minimal hitting sets of the MCSes. In next section, we outline the axiom pinpointing methods for $\mathcal{EL}$ family of description logics.

3 Axiom pinpointing in $\mathcal{EL}$ Family of DLs

Axiom pinpointing is the problem of explaining unintended subsumption relations in description logics [32,27,23,5,8,16,37,33,9,34,6,24,35]. The goal of axiom pinpointing is to find minimal explanations for unintended subsumption relations. Throughout this paper, the following standard definition of MinA (and of nMinA) [34] is assumed.

Definition 4 (nMinA/MinA). *Let $\mathcal{O}$ be an $\mathcal{EL}^+$ ontology and $C, D \in \mathsf{N_C}^\top$ be concept names with $C \sqsubseteq_\mathcal{O} D$. Let $\mathcal{S}$ be a subset of $\mathcal{O}$ such that $C \sqsubseteq_\mathcal{S} D$. If $\mathcal{S}$ is such that $C \sqsubseteq_\mathcal{S} D$ and $C \not\sqsubseteq_{\mathcal{S}'} D$ for $\mathcal{S}' \subset \mathcal{S}$, then $\mathcal{S}$ is a* minimal axiom set *(MinA) w.r.t. $C \sqsubseteq_\mathcal{O} D$. Otherwise, $\mathcal{S}$ is a non-minimal axiom set (nMinA) w.r.t. $C \sqsubseteq_\mathcal{O} D$.*

For axiom pinpointing in the $\mathcal{EL}$ family of DLs, two main approaches have been proposed [5,8,34,35]. Former approach uses the labeled classification method [5,7,8] that constitutes a pinpointing propositional formula $\Phi_{C \sqsubseteq_\mathcal{O} D}$ for any subsumption relation $C \sqsubseteq_\mathcal{O} D$ w.r.t. ontology $\mathcal{O}$ such that the minimal models of $\Phi_{C \sqsubseteq_\mathcal{O} D}$ coincide with the minimal axiom sets (MinAs). The main drawback of this method is that the pinpointing formula $\Phi_{C \sqsubseteq_\mathcal{O} D}$ is worst-case exponential on the size of $\mathcal{O}$ and enumeration of minimal models is NP-hard. In contrast, more recent work [34,35] showed that concept subsumption can be encoded to a Horn formula[2], and that the axiom pinpointing problem can be solved on this Horn formula [34,35].

[2] Subsumption in the description logic $\mathcal{HL}$ with GCIs is basically the implication problem for the propositional Horn clauses, which can be solved in polynomial time [8].

3.1 Propositional Horn Encoding

For the $\mathcal{EL}$ family of DLs, the propositional encoding [34,35] generates a definite Horn formula that encodes the classification of any ontology $\mathcal{O}$. The encoding assigns a unique fresh variable $x_{[C]}$ to every concept description $C \in \mathcal{CS}$ in a normalized ontology $\mathcal{O}$. Every concept assertion in $\mathcal{A} := \{a \in \mathcal{O} | a \text{ is a GCI}\}$ is translated into a definite Horn clause of the propositional formula $\mathcal{F}$ using the mapping function $\gamma : \mathcal{A} \mapsto \mathcal{F}$ defined in Table 4. The CNF propositional formula $\phi_{\mathcal{O}}$ is defined as:

$$\phi_{\mathcal{O}} := \bigwedge_{a \in \mathcal{A}} \gamma(a)$$

In an $\mathcal{EL}$ ontology $\mathcal{O}$, the subsumption problem $C \sqsubseteq_{\mathcal{O}} D$ is equivalent to checking the unsatisfiability of $\phi_{\mathcal{O}} \cup \{x_{[C]}, \neg x_{[D]}\}$.

Theorem 3 (Theorem 1 in [35]). *Let $\mathcal{O}$ is a normalized $\mathcal{EL}$ ontology and $C, D \in \mathsf{N_C}^{\top}$ are concept names. The subsumption relation $C \sqsubseteq_{\mathcal{O}} D$ holds if and only if propositional formula $\phi_{\mathcal{O}} \wedge x_{[C]} \wedge \neg x_{[D]}$ is unsatisfiable.*

$\gamma : \mathcal{A} \mapsto \mathcal{F}$
$C_1 \sqcap \cdots \sqcap C_k \sqsubseteq D \mapsto \neg x_{[C_1]} \vee \cdots \vee \neg x_{[C_k]} \vee x_{[D]}$
$C \sqsubseteq \exists r.D \mapsto \neg x_{[C]} \vee x_{[\exists r.D]}$
$\exists r.C \sqsubseteq D \mapsto \neg x_{[\exists r.C]} \vee x_{[D]}$
$C \sqsubseteq C \mapsto \neg x_{[C]} \vee x_{[C]}$
$C \sqsubseteq \top \mapsto \neg x_{[C]} \vee \texttt{true}$

Table 4: $\mathcal{EL}$ to SAT mapping function.

For an $\mathcal{EL}^+$ ontology, the encoding method described in [35] assign a selection variable (i.e., s_{a_i}) to every assertion axiom (i.e., a_i) in $\mathcal{A}$ and translate the classification rules of Table 3 into a horn propositional formula:

$$\phi_{\mathcal{O}_{so}} := \bigwedge_{a \in \mathcal{A}} \{s_{[a]} \rightarrow \gamma(a)\}$$

$$\phi^{\text{all}}_{\mathcal{O}(\text{po})} := \{s_{[a_i]} \wedge s_{[a_j]} \rightarrow s_{[a_k]} | a_i \in \mathcal{A} \, , \, a_j \in \mathcal{O}\}$$

where a_k is the entailed assertion obtained by applying any completion rule (i.e, $\text{CR}_0, \ldots, \text{CR}_5$) of Table 3. The following Horn propositional formula $\phi^{\text{all}}_{\mathcal{O}}$ represents the complete classification of any normalized $\mathcal{EL}^+$ ontology $\mathcal{O}$.

$$\phi^{\text{all}}_{\mathcal{O}} := \phi_{\mathcal{O}_{so}} \wedge \phi^{\text{all}}_{\mathcal{O}(\text{po})}$$

The following theorem is fundamental for earlier work [34,35], and is extended in the next section to relate MinAs with MUSes of propositional formulae.

Theorem 4 (Theorem 3 in [35]). *Given an $\mathcal{EL}^+$ normalized ontology $\mathcal{O}$, for every $\mathcal{S} \subseteq \mathcal{T}$ and for every pair of concept names $C, D \in \mathsf{N_C}^{\top}$, $C \sqsubseteq_{\mathcal{S}} D$ if and only if the Horn propositional formula,*

$$\phi^{\text{all}}_{\mathcal{O}(\text{po})} \wedge_{ax_i \in \mathcal{S}} (s_{[ax_i]}) \wedge (\neg s_{[C \sqsubseteq D]})$$

is unsatisfiable.

The dedicated axiom pinpointing algorithms [34,35] exploit the ideas from early work on SAT solving [39,26] and AllSMT [18], compute MinAs for any (unintended) subsumption relation in an $\mathcal{EL}^+$ ontology. Although effective at computing minimal axiom sets (MinAs), the SMT based algorithm often fails to enumerate all MinAs to completion, or proving that no additional MinA exists. In order to resolve these inefficiencies, we propose an axiom pinpointing algorithm (Section 4) that exploits MaxSAT-based MCS enumeration [20] for computing MinAs. The proposed solution is able to compute the MinAs for the vast majority of existing problem instances in medical ontologies (e.g. Gene [1], GALEN [30], NCI [36] and SNOMED-CT [38]) with many orders of magnitude performance improvements over the previous work [34,35]. Thus, achieving a comprehensive performance gains over the earlier work [34,35,7,8].

4 Axiom Pinpointing with MUS Extraction

This section shows that the computation of MinAs can be related with MUS enumeration of the Horn formula $\phi^{\text{all}}_{\mathcal{O}(\text{po})}$. We also explain the axiom pinpointing algorithm that exploits MaxSAT-based MCS enumeration [20] for computing MinAs in description logic $\mathcal{EL}^+$. We argue that the existing approaches for MUS enumeration [20,25,22] (Section 2.3) can used to compute the MinAs for an (unintended) subsumption relation, and concludes summarizing the EL2MCS tool proposed in this paper.

4.1 MinAs as MUSes

Although not explicitly stated, the relation between axiom pinpointing and MUS extraction has been apparent in earlier work [8,34,35]. Indeed, from the encoding of axiom pinpointing to Horn formula, it is immediate the following result:

Theorem 5. *Given an $\mathcal{EL}^+$ Ontology $\mathcal{O}$, for every $\mathcal{S} \subseteq \mathcal{O}$ and for every pair of concept names $C, D \in N_C^\top$, $\mathcal{S}$ is a MinA of $C \sqsubseteq_\mathcal{S} D$ if and only if the Horn propositional formula,*

$$\phi^{\text{all}}_{\mathcal{O}(\text{po})} \wedge_{ax_i \in \mathcal{S}} (s_{[ax_i]}) \wedge (\neg s_{[C \sqsubseteq D]}) \tag{1}$$

is minimally unsatisfiable.

Proof. By Theorem 4, $C \sqsubseteq_\mathcal{S} D$ if and only if the associated Horn formula (1) is unsatisfiable. For a MinA $\mathcal{S} \subseteq \mathcal{O}$, minimal unsatisfiability of (1) (with $\mathcal{O}$ replaced by $\mathcal{S}$) results from the MinA computation algorithm proposed in earlier work [34,35]. □

Based on Theorem 5 and the MUS enumeration approaches summarized in Section 2.3, we outline our axiom pinpointing approach (Section 4.2) that exploits the idea of MaxSAT-based MCS enumeration [20,25].

4.2 MCS Enumeration using MaxSAT

Earlier work [34,35] explicitly enumerates the selection variables (i.e., $s_{[ax_i]}$) in a AllSMT-inspired approach [18]. In contrast, our approach is to model the problem as partial maximum satisfiability (MaxSAT), and enumerate the MUSes of the MaxSAT problem formulation. All clauses in $\phi^{\text{all}}_{\mathcal{O}(\text{po})}$ are declared as hard clauses, i.e. they must be satisfied. Observe that, by construction, $\phi^{\text{all}}_{\mathcal{O}(\text{po})}$ is satisfiable. In addition, the constraint $C \sqsubseteq_{\mathcal{O}} D$ is encoded with another hard clause, namely $(\neg s_{[C \sqsubseteq_{\mathcal{O}} D]})$. Finally, the variable associated with each axiom ax_i, $s_{[ax_i]}$ denotes a *unit soft clause*. The intuitive justification is that the goal is to include as many axioms as possible, leaving out a minimal set which if included will cause the complete formula to be unsatisfiable. Thus, each of these sets represents an MCS of the MaxSAT problem formulation, but also a minimal set of axioms that needs to be dropped for the subsumption relation not to hold. MCS enumeration can easily be implemented with a MaxSAT solver [20,25] or with a dedicated algorithm [22]. Moreover, we can now use minimal hitting set dualization [31,12,10,20] to obtain the MUSes we are looking for, starting from the previously computed MCSes. This is the approach implemented in this paper. One problem with this approach is that all MCSes need to be enumerated *before* the first MUS is computed. As a result, more recent work proposed alternative approaches which enable both MUSes and MCSes to be computed while enumeration of MUSes (and MCSes) takes place [19,29]. Nevertheless, if the number of MCSes is manageable, earlier work is expected to be more efficient [20]. It should be noted that, all existing approaches for MUS enumeration relate with Theorem 2, in that enumeration of MUSes is achieved by explicitly or implicitly computing the minimal hitting sets of all MCSes. Regarding MCS enumeration, two main approaches have been studied. One exploits MaxSAT enumeration [20,25]. More recent work proposes dedicated MCS extraction algorithms, also capable of enumerating MCSes [22]. The approach proposed in this paper exploits MaxSAT enumeration.

4.3 The EL2MCS Tool

This section summarizes the organization of the EL2MCS tool, that exploits MUS enumeration for axiom pinpointing. The organization of EL2MCS is shown in Figure 1. The first step is similar to EL2SAT [34,35] in that a propositional Horn formula is generated. The next step, however, exploits the ideas in the previous section, and generates a partial MaxSAT encoding. As outlined earlier, we can now enumerate the MCSes of the partial MaxSAT formula. This is achieved with the CAMUS2 tool [22][3]. The final step is to exploit minimal hitting set dualization for computing all the MUSes given the set of MCSes [20]. This is achieved with the CAMUS tool[4]. It should be observed that, although MCS enumeration uses CAMUS2 (a modern implementation of the MCS enumerator in CAMUS [20], capable of handling partial MaxSAT formulae), alternative MCS enumeration approaches were considered [22], not being as efficient.

5 Experimental Results

The experiments were performed on an HPC cluster, with dual quad-core Intel Xeon 2GHz processors, with 32GB of physical memory. The tools CEL, EL2SAT and EL2MCS

[3] Available from http://logos.ucd.ie/web/doku.php?id=mcsls.
[4] Available from http://sun.iwu.edu/~mliffito/camus/.

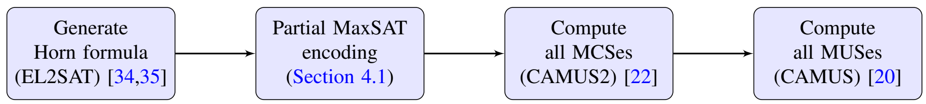

Fig. 1: The EL2MCS tool

were given a timeout of 3600 seconds and a memory limit of 16GB. EL2MCS uses the partial MaxSAT instances generated with the help of EL2SAT Horn propositional encoding tool[5]. The medical ontologies used in the experimentation are GALEN [30], Gene [1], NCI [36] and SNOMED-CT [38][6]. We have used the 450 subsumption query instances which are studied in earlier work [34,35] (and which are available from the EL2SAT website). Moreover, unless otherwise stated, the experiments consider the optimizations proposed in earlier work [34,35], and exploited in the EL2SAT tool. EL2MCS and EL2SAT use the cone-of-influence reduction technique[35], which reduces the size of propositional satisfiability instances significantly. CEL can only be instructed to compute 10 MinAs, and then the tool terminates. To our best knowledge, this is motivated by performance issues with the approach used in CEL. Table 5 summarizes the statistics of running CEL, EL2SAT and EL2MCS. EL2MCS outperforms EL2SAT on 74.9% of the instances. EL2MCS outperforms CEL on 82% of instances. EL2SAT does not terminate on 207 instances. The statistics further support the conclusion that extraction of MUSes provides a far more robust solution than proposed in [34,35]. Given its built-in limitation, CEL never attempts to prove the non-existence of additional MinAs. Thus, the information about "solved" instances is meant to signify that the total number of MUSes was computed, even though the tool *did not* prove the non-existence of additional MinAs. As a result, the comparison between CEL and EL2MCS excludes the problems instances with more than 10 MUSes/MinAs, for which EL2MCS takes more time than CEL. From Table 5, the following conclusions can be drawn. First, EL2MCS achieves clear performance gains over the two other tools, and this while guaranteeing that there are no more MUSes/MinAs. Second, given that most instances have less than 10 MinAs, CEL terminates on more instances than the ones solved by EL2SAT. However, CEL is not proving there are no more MinAs, and in several cases EL2SAT proves non-existence of more MinAs. Table 6 shows that EL2MCS computes more MUSes (equivalently MinAs) than either EL2SAT or CEL. Observe that EL2SAT is quite efficient to find most MUSes when COI reduction is applied but is often unable to prove the non-existence of additional MUSes as shown in Table 5. The Figure 2 shows scatter plots comparing the solving times of EL2MCS and EL2SAT, as well as the termination time of CEL against the solving time of EL2MCS. As can be concluded, the performance difference is conclusive in both cases.

[5] The $\mathcal{EL}^+$ to SAT encoder is denoted el2sat_all, whereas the axiom pinpointing tool is el2sat_all_mins. These tools are available from http://disi.unitn.it/~rseba/elsat/. Moreover, this site also contains the subsumption query instances used in the experiments.

[6] GENE, GALEN and NCI ontologies are freely available at http://lat.inf.tu-dresden.de/~meng/toyont.html. The SNOMED-CT ontology was requested from IHTSDO under a nondisclosure license agreement.

	EL2SAT	EL2MCS
# Solved	241	448
% Solved	53.6 %	99.6 %
# Wins	72	339
% Wins	16.0%	74.9%

(a) EL2SAT vs. EL2MCS

	CEL	EL2MCS
# Terminated	337	448
% Terminated	74.9 %	99.6%
# Wins	19	369
% Wins	0.4 %	82 %

(b) CEL vs. EL2MCS

Table 5: Statistics summarizing the results on a universe of 450 problem instances with a timeout of 3600s, with COI reduction.

EL2SAT	EL2MCS	Ties
2	19	429

(a) EL2SAT vs. EL2MCS

CEL	EL2MCS	Ties
0	75	375

(b) CEL vs. EL2MCS

Table 6: Number of times a tool computes more MUSes than the other within a timeout of 3600s, again with COI reduction.

6 Conclusions & Future Work

The main contribution of this paper is to relate axiom pinpointing with MUS extraction. As a result, this enables exploiting different MUS extraction and enumeration algorithms for the problem of axiom pinpointing. Preliminary results, obtained using off-the-shelf tools, show categorical performance gains over the current state of the art in $\mathcal{EL}^+$ axiom pinpointing. The results substantiate a further analysis of the uses of MUSes and MCSes in axiom pinpointing for the $\mathcal{EL}$ family of DLs.

Acknowledgement. We thank the authors of EL2SAT for authorizing the use of the most recent, yet unpublished, version of their work. This work is partially supported by SFI PI grant BEACON (09/IN.1/I2618), by FCT grant POLARIS (PTDC/EIA-CCO/-123051/2010), and by national funds through FCT with reference UID/CEC/50021/2013.

References

1. Michael Ashburner, Catherine A. Ball, Judith A. Blake, David Botstein, Heather Butler, J. Michael Cherry, Allan P. Davis, Kara Dolinski, Selina S. Dwight, Janan T. Eppig, and et al. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.
2. F. Baader, C. Lutz, and B. Suntisrivaraporn. Efficient reasoning in $\mathcal{EL}^+$. In *DL*, 2006.
3. Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the $\mathcal{EL}$ envelope. In *IJCAI*, pages 364–369, 2005.
4. Franz Baader and Bernhard Hollunder. Embedding defaults into terminological knowledge representation formalisms. *J. Autom. Reasoning*, 14(1):149–180, 1995.
5. Franz Baader, Carsten Lutz, and Boontawee Suntisrivaraporn. CEL - A polynomial-time reasoner for life science ontologies. In *IJCAR*, pages 287–291, 2006.

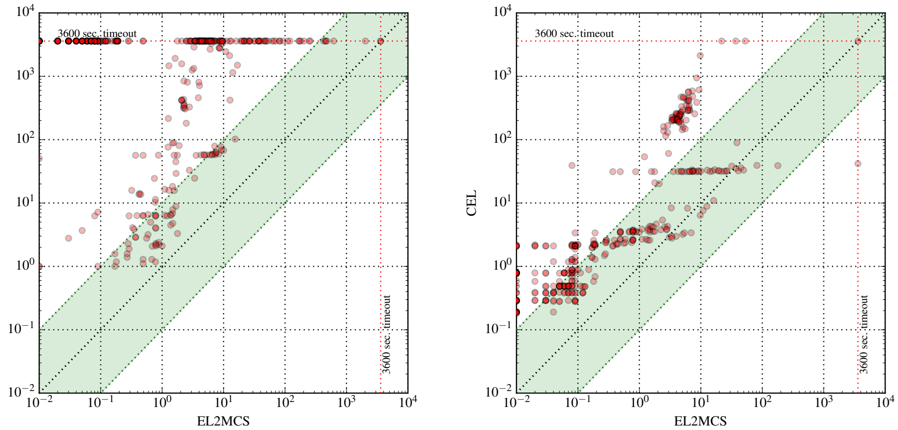

Fig. 2: Scatter plots comparing EL2MCS with EL2SAT and CEL, on all problem instances, with COI reduction

6. Franz Baader and Rafael Peñaloza. Axiom pinpointing in general tableaux. *J. Log. Comput.*, 20(1):5–34, 2010.
7. Franz Baader, Rafael Peñaloza, and Boontawee Suntisrivaraporn. Pinpointing in the description logic $\mathcal{EL}$. In *DL*, 2007.
8. Franz Baader, Rafael Peñaloza, and Boontawee Suntisrivaraporn. Pinpointing in the description logic $\mathcal{EL}^+$. In *KI*, pages 52–67, 2007.
9. Franz Baader and Boontawee Suntisrivaraporn. Debugging SNOMED CT using axiom pinpointing in the description logic $\mathcal{EL}^+$. In *KR-MED*, 2008.
10. James Bailey and Peter J. Stuckey. Discovery of minimal unsatisfiable subsets of constraints using hitting set dualization. In *PADL*, pages 174–186, 2005.
11. Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors. *Handbook of Satisfiability*, volume 185, 2009.
12. Elazar Birnbaum and Eliezer L. Lozinskii. Consistent subsets of inconsistent systems: structure and behaviour. *J. Exp. Theor. Artif. Intell.*, 15(1):25–46, 2003.
13. Sebastian Brandt. Polynomial time reasoning in a description logic with existential restrictions, GCI axioms, and - what else? In *ECAI*, pages 298–302, 2004.
14. Edmund M. Clarke, Jr., Orna Grumberg, and Doron A. Peled. *Model Checking*. MIT Press, Cambridge, MA, USA, 1999.
15. William F. Dowling and Jean H. Gallier. Linear-time algorithms for testing the satisfiability of propositional Horn formulae. *J. Log. Program.*, 1(3):267–284, 1984.
16. Aditya Kalyanpur, Bijan Parsia, Matthew Horridge, and Evren Sirin. Finding All Justifications of OWL DL Entailments. In *ISWC*, pages 267–280, 2007.
17. Boris Konev, Michel Ludwig, Dirk Walther, and Frank Wolter. The logical difference for the lightweight description logic EL. *J. Artif. Intell. Res. (JAIR)*, 44:633–708, 2012.
18. Shuvendu K. Lahiri, Robert Nieuwenhuis, and Albert Oliveras. SMT techniques for fast predicate abstraction. In *CAV*, pages 424–437, 2006.
19. Mark H. Liffiton and Ammar Malik. Enumerating Infeasibility: Finding Multiple MUSes Quickly. In *CPAIOR*, pages 160–175, 2013.
20. Mark H. Liffiton and Karem A. Sakallah. Algorithms for computing minimal unsatisfiable subsets of constraints. *J. Autom. Reasoning*, 40(1):1–33, 2008.

21. Carsten Lutz, David Toman, and Frank Wolter. Conjunctive query answering in the description logic EL using a relational database system. In *IJCAI*, pages 2070–2075, 2009.
22. Joao Marques-Silva, Federico Heras, Mikolás Janota, Alessandro Previti, and Anton Belov. On computing minimal correction subsets. In *IJCAI*, 2013.
23. Thomas Andreas Meyer, Kevin Lee, Richard Booth, and Jeff Z. Pan. Finding maximally satisfiable terminologies for the description logic $\mathcal{ALC}$. In *AAAI*, pages 269–274, 2006.
24. Kodylan Moodley, Thomas Meyer, and Ivan José Varzinczak. Root justifications for ontology repair. In *RR*, pages 275–280, 2011.
25. António Morgado, Mark H. Liffiton, and Joao Marques-Silva. MaxSAT-based MCS enumeration. In *HVC*, pages 86–101, 2012.
26. Matthew W. Moskewicz, Conor F. Madigan, Ying Zhao, Lintao Zhang, and Sharad Malik. Chaff: Engineering an efficient sat solver. In *DAC*, pages 530–535, 2001.
27. Bijan Parsia, Evren Sirin, and Aditya Kalyanpur. Debugging OWL ontologies. In *WWW*, pages 633–640, 2005.
28. Rafael Peñaloza and Baris Sertkaya. On the complexity of axiom pinpointing in the EL family of description logics. In *KR*, pages 280–289, 2010.
29. Alessandro Previti and Joao Marques-Silva. Partial MUS enumeration. In *AAAI*, 2013.
30. Alan L. Rector and Ian R. Horrocks. Experience building a large, re-usable medical ontology using a description logic with transitivity and concept inclusions. In *Workshop on Ontological Engineering*, pages 414–418, 1997.
31. Raymond Reiter. A theory of diagnosis from first principles. *Artif. Intell.*, 32(1):57–95, 1987.
32. Stefan Schlobach and Ronald Cornet. Non-standard reasoning services for the debugging of description logic terminologies. In *IJCAI*, pages 355–362, 2003.
33. Stefan Schlobach, Zhisheng Huang, Ronald Cornet, and Frank van Harmelen. Debugging incoherent terminologies. *J. Autom. Reasoning*, 39(3):317–349, 2007.
34. Roberto Sebastiani and Michele Vescovi. Axiom pinpointing in lightweight description logics via Horn-SAT encoding and conflict analysis. In *CADE*, pages 84–99, 2009.
35. Roberto Sebastiani and Michele Vescovi. Debugging $\mathcal{EL}^+$ ontologies via SAT and SMT techniques. *Journal of Artificial Intelligence Research (submitted)*, 2014. Available from http://disi.unitn.it/~rseba/elsat/elsat_jair14_submitted.pdf.
36. Nicholas Sioutos, Sherri de Coronado, Margaret W. Haber, Frank W. Hartel, Wen-Ling Shaiu, and Lawrence W. Wright. NCI thesaurus: A semantic model integrating cancer-related clinical and molecular information. *Journal of Biomedical Informatics*, 40(1):30–43, 2007.
37. Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, and Yarden Katz. Pellet: A practical OWL-DL reasoner. *J. Web Sem.*, 5(2):51–53, 2007.
38. Kent A. Spackman, Keith E. Campbell, and Roger A. Côté. SNOMED RT: a reference terminology for health care. In *AMIA*, 1997.
39. Lintao Zhang, Conor F. Madigan, Matthew W. Moskewicz, and Sharad Malik. Efficient Conflict Driven Learning in Boolean Satisfiability Solver. In *ICCAD*, pages 279–285, 2001.