The Role of Unsatisfiable Boolean Constraints in Lightweight Description Logics

Doctoral Thesis Defense

19 July 2016

M. Fareed Arif

Advisor: Joao-Marques Silva, Alexey Lastovetsky

Contents

What? - Problem

Why? - Relevance

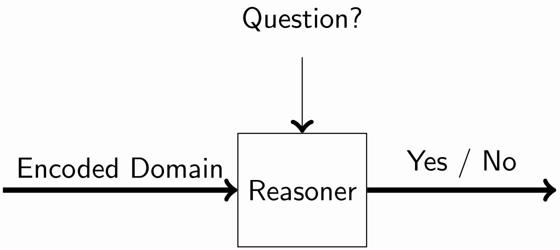
How? - Solution

Results

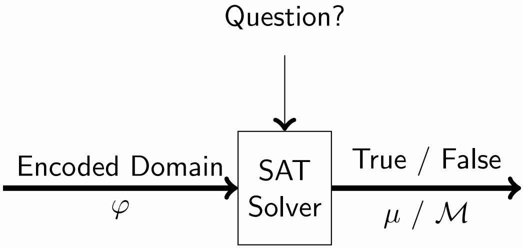
Conclusions

Publications

Problem?



Problem? - Boolean Satisfiability Problem (SAT)



Boolean Satisfiability Problem (SAT)

A literal is a variable or its negation

- Clause: A **disjunction** of literals

$$(c \vee \neg a)$$

  Satisfied clause: at least one literal is true under the given assignment to variables

- CNF: A **conjunction** of clauses φ

$$(c \vee b) \wedge (c \vee \neg a)$$

  Satisfied CNF: all of its clauses are true under the given assignment (μ) to variables

Boolean Satisfiability Problem (SAT) (cont.)

- MUS: An irreducible unsatisfiable set of clauses $\mathcal{M}$ from φ

$$\varphi = \overbrace{(a)}^{C_1} \wedge \overbrace{(\neg a)}^{C_2} \wedge \overbrace{(\neg a \vee b)}^{C_3} \wedge \overbrace{(\neg b)}^{C_4} \wedge \overbrace{(\neg a \vee c)}^{C_5} \wedge \overbrace{(\neg c)}^{C_6}$$

- MCS: irreducible set of clauses such that complement (MSS) is satisfiable

$$\varphi = \overbrace{(a)}^{C_1} \wedge \overbrace{(\neg a)}^{C_2} \wedge \overbrace{(\neg a \vee b)}^{C_3} \wedge \overbrace{(\neg b)}^{C_4} \wedge \overbrace{(\neg a \vee c)}^{C_5} \wedge \overbrace{(\neg c)}^{C_6}$$

Boolean Satisfiability Problem (SAT) (cont.)

- MUS: An irreducible unsatisfiable set of clauses $\mathcal{M}$ from φ

$$\varphi = \overbrace{\boxed{(a)}}^{C_1} \wedge \overbrace{\boxed{(\neg a)}}^{C_2} \wedge \overbrace{(\neg a \vee b)}^{C_3} \wedge \overbrace{(\neg b)}^{C_4} \wedge \overbrace{(\neg a \vee c)}^{C_5} \wedge \overbrace{(\neg c)}^{C_6}$$

$$\texttt{MUSes} = \{\{C_1, C_2\}\}$$

- MCS: irreducible set of clauses such that complement (MSS) is satisfiable

$$\varphi = \overbrace{\boxed{(a)}}^{C_1} \wedge \overbrace{(\neg a)}^{C_2} \wedge \overbrace{(\neg a \vee b)}^{C_3} \wedge \overbrace{(\neg b)}^{C_4} \wedge \overbrace{(\neg a \vee c)}^{C_5} \wedge \overbrace{(\neg c)}^{C_6}$$

$$\texttt{MCSes} = \{\{C_1\}\}$$

Boolean Satisfiability Problem (SAT) (cont.)

- MUS: An irreducible unsatisfiable set of clauses $\mathcal{M}$ from φ

$$\varphi = \overbrace{\boxed{(a)}}^{C_1} \wedge \overbrace{(\neg a)}^{C_2} \wedge \overbrace{\boxed{(\neg a \vee b)}}^{C_3} \wedge \overbrace{\boxed{(\neg b)}}^{C_4} \wedge \overbrace{(\neg a \vee c)}^{C_5} \wedge \overbrace{(\neg c)}^{C_6}$$

$$\texttt{MUSes} = \{\{C_1, C_2\}, \{C_1, C_3, C_4\}\}$$

- MCS: irreducible set of clauses such that complement (MSS) is satisfiable

$$\varphi = \overbrace{(a)}^{C_1} \wedge \overbrace{\boxed{(\neg a)}}^{C_2} \wedge \overbrace{\boxed{(\neg a \vee b)}}^{C_3} \wedge \overbrace{(\neg b)}^{C_4} \wedge \overbrace{\boxed{(\neg a \vee c)}}^{C_5} \wedge \overbrace{(\neg c)}^{C_6}$$

$$\texttt{MCSes} = \{\{C_1\}, \{C_2, C_3, C_5\}\}$$

Boolean Satisfiability Problem (SAT) (cont.)

- MUS: An irreducible unsatisfiable set of clauses $\mathcal{M}$ from φ

$$\varphi = \overbrace{(a)}^{C_1} \wedge \overbrace{(\neg a)}^{C_2} \wedge \overbrace{(\neg a \vee b)}^{C_3} \wedge \overbrace{(\neg b)}^{C_4} \wedge \overbrace{(\neg a \vee c)}^{C_5} \wedge \overbrace{(\neg c)}^{C_6}$$

$$\texttt{MUSes} = \{\{C_1, C_2\}, \{C_1, C_3, C_4\}, \{C_1, C_5, C_6\}\}$$

- MCS: irreducible set of clauses such that complement (MSS) is satisfiable

$$\varphi = \overbrace{(a)}^{C_1} \wedge \overbrace{(\neg a)}^{C_2} \wedge \overbrace{(\neg a \vee b)}^{C_3} \wedge \overbrace{(\neg b)}^{C_4} \wedge \overbrace{(\neg a \vee c)}^{C_5} \wedge \overbrace{(\neg c)}^{C_6}$$

$$\texttt{MCSes} = \{\{C_1\}, \{C_2, C_3, C_5\}, \{C_2, C_3, C_6\}\}$$

Boolean Satisfiability Problem (SAT) (cont.)

- MUS: An irreducible unsatisfiable set of clauses $\mathcal{M}$ from φ

$$\varphi = \overbrace{\boxed{(a)}}^{C_1} \wedge \overbrace{(\neg a)}^{C_2} \wedge \overbrace{(\neg a \vee b)}^{C_3} \wedge \overbrace{(\neg b)}^{C_4} \wedge \overbrace{\boxed{(\neg a \vee c)}}^{C_5} \wedge \overbrace{\boxed{(\neg c)}}^{C_6}$$

$$\texttt{MUSes} = \{\{C_1, C_2\}, \{C_1, C_3, C_4\}, \{C_1, C_5, C_6\}\}$$

- MCS: irreducible set of clauses such that complement (MSS) is satisfiable

$$\varphi = \overbrace{(a)}^{C_1} \wedge \overbrace{\boxed{(\neg a)}}^{C_2} \wedge \overbrace{(\neg a \vee b)}^{C_3} \wedge \overbrace{\boxed{(\neg b)}}^{C_4} \wedge \overbrace{\boxed{(\neg a \vee c)}}^{C_5} \wedge \overbrace{(\neg c)}^{C_6}$$

$$\texttt{MCSes} = \{\{C_1\}, \{C_2, C_3, C_5\}, \{C_2, C_3, C_6\}, \{C_2, C_4, C_5\}\}$$

Boolean Satisfiability Problem (SAT) (cont.)

- MUS: An irreducible unsatisfiable set of clauses $\mathcal{M}$ from φ

$$\varphi = \overbrace{\boxed{(a)}}^{C_1} \wedge \overbrace{(\neg a)}^{C_2} \wedge \overbrace{(\neg a \vee b)}^{C_3} \wedge \overbrace{(\neg b)}^{C_4} \wedge \overbrace{\boxed{(\neg a \vee c)}}^{C_5} \wedge \overbrace{\boxed{(\neg c)}}^{C_6}$$

$$\texttt{MUSes} = \{\{C_1, C_2\}, \{C_1, C_3, C_4\}, \{C_1, C_5, C_6\}\}$$

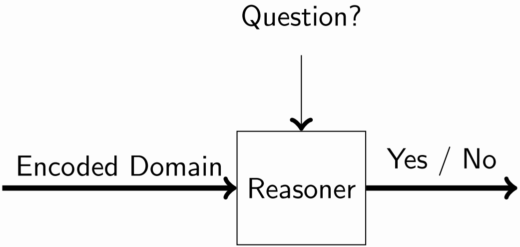
- MCS: irreducible set of clauses such that complement (MSS) is satisfiable

$$\varphi = \overbrace{(a)}^{C_1} \wedge \overbrace{\boxed{(\neg a)}}^{C_2} \wedge \overbrace{(\neg a \vee b)}^{C_3} \wedge \overbrace{\boxed{(\neg b)}}^{C_4} \wedge \overbrace{(\neg a \vee c)}^{C_5} \wedge \overbrace{\boxed{(\neg c)}}^{C_6}$$

$$\texttt{MCSes} = \{\{C_1\}, \{C_2, C_3, C_5\}, \{C_2, C_3, C_6\}, \{C_2, C_4, C_5\}, \{C_2, C_4, C_6\}\}$$

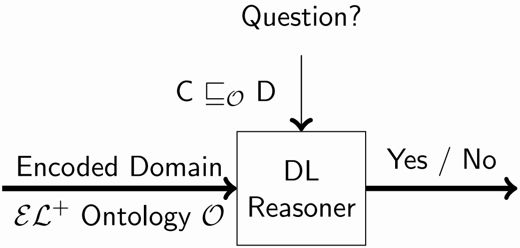
- Hitting set duality:
  - An MUS of φ is an irreducible hitting set of MCSes
  - An MCS of φ is an irreducible hitting set of MUSes

Problem?



Problem? - Lightweight Description Logic $\mathcal{EL}^+$

- Description Logics (DLs) are formal knowledge representation languages (more expressive than propositional logic)



- Lightweight Description Logic $\mathcal{EL}^+$ is tractable and has efficient polynomial-time reasoning services.

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

- N_C, N_R denote concept & role names, respectively
- Concept descriptions formed using 3 constructors below
- Ontology $\mathcal{O}$ is a finite set of GCI's and RI's

	Syntax	Semantics
top	$\top$	$\Delta^{\mathcal{I}}$
conjunction	$X \sqcap Y$	$X^{\mathcal{I}} \cap Y^{\mathcal{I}}$
existential restriction	$\exists r.X$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}} : (x, y) \in r^{\mathcal{I}} \wedge y \in X^{\mathcal{I}}\}$
general concept inclusion	$X \sqsubseteq Y$	$X^{\mathcal{I}} \subseteq Y^{\mathcal{I}}$
role inclusion	$r_1 \circ \cdots \circ r_n \sqsubseteq s$	$r_1^{\mathcal{I}} \circ \cdots \circ r_n^{\mathcal{I}} \subseteq s^{\mathcal{I}}$

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

- N_C, N_R denote concept & role names, respectively
- Concept descriptions formed using 3 constructors below
- Ontology $\mathcal{O}$ is a finite set of GCI's and RI's
- Interpretation: $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where
  - $\Delta^{\mathcal{I}}$ is non-empty set of individuals, and
  - $\cdot^{\mathcal{I}}$ maps
    - Each $C \in N_C$ to $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
    - Each $r \in N_R$ to a binary relation $r^{\mathcal{I}}$ in $\Delta^{\mathcal{I}}$
  - $\cdot^{\mathcal{I}}$ is defined inductively for arbitrary concept descriptions:

	Syntax	Semantics
top	$\top$	$\Delta^{\mathcal{I}}$
conjunction	$X \sqcap Y$	$X^{\mathcal{I}} \cap Y^{\mathcal{I}}$
existential restriction	$\exists r.X$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}} : (x, y) \in r^{\mathcal{I}} \wedge y \in X^{\mathcal{I}}\}$
general concept inclusion	$X \sqsubseteq Y$	$X^{\mathcal{I}} \subseteq Y^{\mathcal{I}}$
role inclusion	$r_1 \circ \cdots \circ r_n \sqsubseteq s$	$r_1^{\mathcal{I}} \circ \cdots \circ r_n^{\mathcal{I}} \subseteq s^{\mathcal{I}}$

Lightweight Description Logic $\mathcal{EL}^+$ (cont.)

- $\mathcal{I}$ is a model of $\mathcal{O}$ if semantics conditions are satisfied for every concept inclusion (GCI or RI)

Lightweight Description Logic $\mathcal{EL}^+$ (cont.)

- $\mathcal{I}$ is a model of $\mathcal{O}$ if semantics conditions are satisfied for every concept inclusion (GCI or RI)
- Concept Subsumption: C is subsumed w.r.t. D, $C \sqsubseteq_{\mathcal{O}} D$, if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for every model $\mathcal{I}$ of $\mathcal{O}$

Lightweight Description Logic $\mathcal{EL}^+$ (cont.)

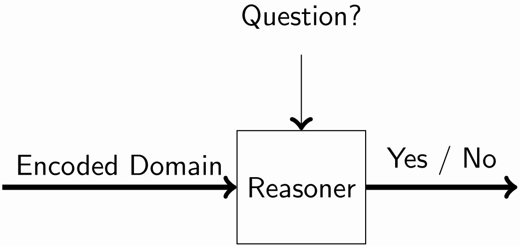
- $\mathcal{I}$ is a model of $\mathcal{O}$ if semantics conditions are satisfied for every concept inclusion (GCI or RI)
- Concept Subsumption: C is subsumed w.r.t. D, $C \sqsubseteq_{\mathcal{O}} D$, if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for every model $\mathcal{I}$ of $\mathcal{O}$
- Classification: infer all subsumption relations between atomic concepts

Example of medical ontology ($\mathcal{O}_{med}$)

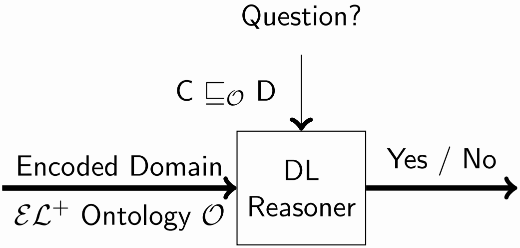
1. Endocarditis $\sqsubseteq$ Inflammation $\sqcap$ $\exists$hasLoc.Endocardium
2. Inflammation $\sqsubseteq$ Disease $\sqcap$ $\exists$actsOn.Tissue
3. Endocardium $\sqsubseteq$ Tissue $\sqcap$ $\exists$contIn.HeartValve
4. HeartValve $\sqsubseteq$ $\exists$contIn.Heart
5. HeartDisease $\equiv$ Disease $\sqcap$ $\exists$hasLoc.Heart
6. contIn $\circ$ contIn $\sqsubseteq$ contIn
7. hasLoc $\circ$ contIn $\sqsubseteq$ hasLoc

- Question? example:
  - Endocarditis $\sqsubseteq_{\mathcal{O}_{med}}$ HeartDisease explained by axioms above

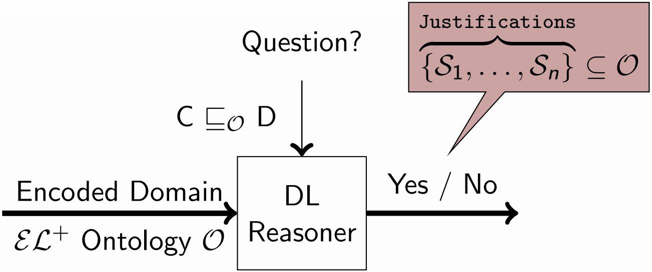
Problem?



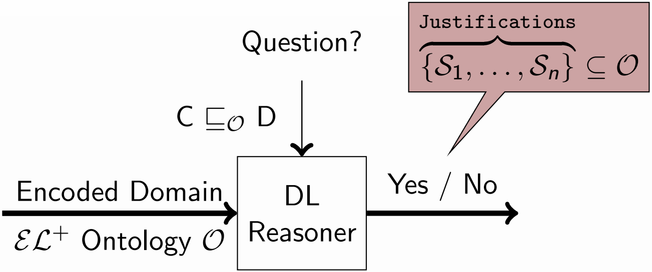
Problem? - Axiom Pinpointing in Lightweight DLs



Problem? - Axiom Pinpointing in Lightweight DLs

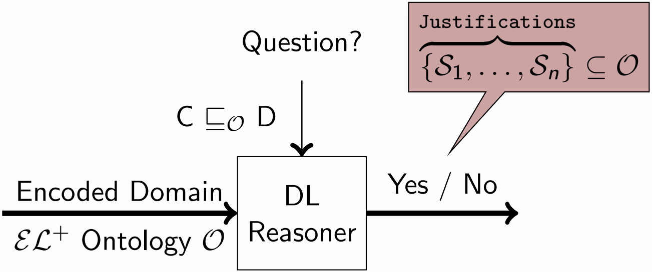


Problem? - Axiom Pinpointing in Lightweight DLs



- Axiom pinpointing: compute minimal set of axioms (**MinAs**) that explain some (unintended) inference (concept subsumption) [SHC03, BPS07]
  - Goal is to compute many, or even all, minimal explanations

Problem? - Axiom Pinpointing in Lightweight DLs



- Axiom pinpointing: compute minimal set of axioms (**MinAs**) that explain some (unintended) inference (concept subsumption) [SHC03, BPS07]
  - Goal is to compute many, or even all, minimal explanations
- Repair complement to Diagnosis: is a maximal subset of axioms from which the inference (concept subsumption) relation does not follow.

Contents

What? - Problem

Why? - Relevance

How? - Solution

Results

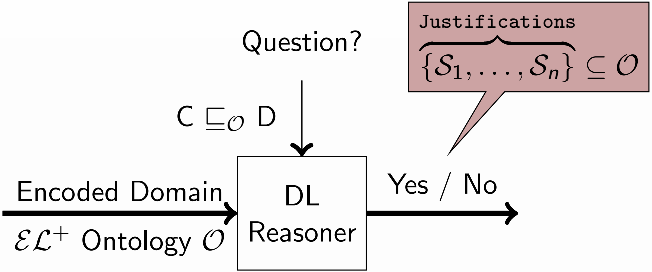
Conclusions

Publications

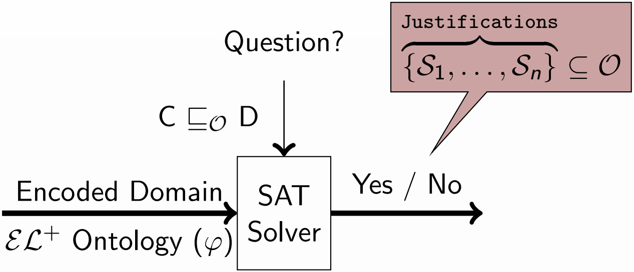
Why? - Relevance of Axiom Pinpointing

- Extensively studied problem [BH95,SC03,MLBP06,BLS06,SCH07,BS08,BP10,...]
- Life sciences ontologies represented with Lightweight DLs ($\mathcal{EL}^+$)
  - E.g. SNOMED CT: Systematized Nomenclature Of Medicine - Clinical Terms $\rightsquigarrow$ 311,000 concepts
- Important Applications:
  - Ontology Debugging and Revision [PSK05,KPSG06,SHCH07]
    - Unintended subsumption in (old version of) SNOMED CT: AmputationOfFinger $\sqsubseteq_{\mathcal{O}_{snomed-ct}}$ AmputationOfHand [BS08]
  - Ontology Matching [JC11,ES13]
  - Explaining Logical Differences [KLW12]
  - Context-based Reasoning [BKP12]
  - Error-tolerant Reasoning [LP14]
  - ...
- Role in SFI Project (BEACON): BoolEAn-based deCision and OptimizatioN procedures for safety or analyzing biological systems.

Sol? - SAT-based axiom pinpointing for $\mathcal{EL}^+$

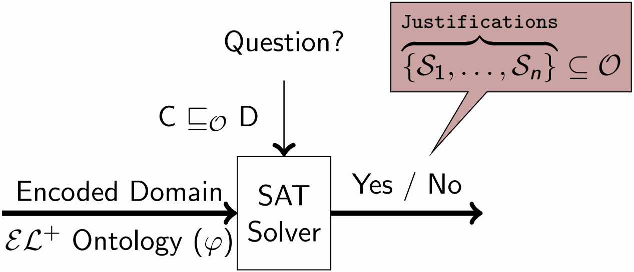


Sol? - SAT-based axiom pinpointing for $\mathcal{EL}^+$



- Axiom pinpointing: compute minimal set of axioms ($\overbrace{\texttt{MinA}\,s}^{\textit{MUS}}$) that explain some (unintended) inference (concept subsumption)
  - Goal is to compute many, or even all, $\overbrace{\texttt{minimal explanations}}^{\texttt{MUSes}}$

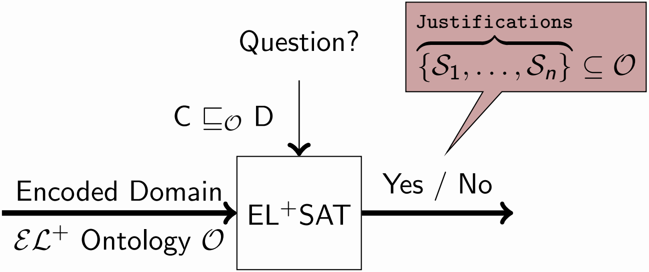
Sol? - SAT-based axiom pinpointing for $\mathcal{EL}^+$



- Axiom pinpointing: compute minimal set of axioms ($\overbrace{\texttt{MinA}\,s}^{MUS}$) that explain some (unintended) inference (concept subsumption)
  - Goal is to compute many, or even all, $\overbrace{\texttt{minimal explanations}}^{\texttt{MUSes}}$
- $\overbrace{\texttt{Repair}}^{\texttt{MSS}}$ complement to $\overbrace{\texttt{Diagnosis}}^{\texttt{MCS}}$: is a maximal subset of axioms from which the inference (concept subsumption) relation does not follow.

SAT-based axiom pinpointing for $\mathcal{EL}^+$ (EL$^+$SAT)

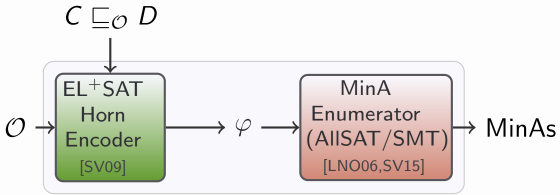
- EL$^+$SAT remained state of the art until 2014 [VS09,VS15]



- **Note**: EL$^+$SAT encodes classification of $\mathcal{O}$ as Horn formula

EL$^+$SAT (cont.)

- EL$^+$SAT main components:
  - Horn Encoder: Encode classification of an $\mathcal{EL}^+$ ontology using a Horn Formula (φ)



  - MinA Enumeration: Enumeration of models inspired by AllSAT/SMT

[LNO06]

Contents

What? - Problem

Why? - Relevance

How? - Solution

Results

Conclusions

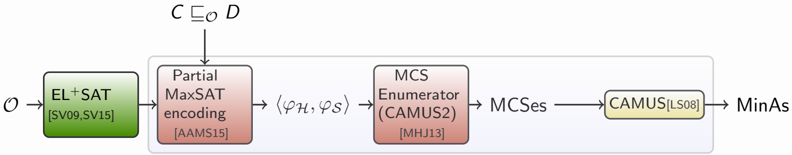
Publications

How? - MUS enumeration of Horn formulae

- For $\mathcal{EL}^+$, a MinA corresponds to an MUS of a Horn formula and the problem of axiom pinpointing corresponds to group MUS enumeration of Horn formulae: [AMMS15]
- How to enumerate MUSes?
  - Use hitting set duality between MUSes and MCSes [R87,BL03]
    - An MUS is an irreducible hitting set of MCSes
    - An MCS is an irreducible hitting set of MUSes
  - We propose state-of-the art axiom pinpointing and debugging tools (use the enumeration of MUSes):
    - **EL2MCS** [AMMS-KI15]
    - **EL2MUS** (core HgMUS) [AMMS-SAT15]
    - **BEACON** [AMMS-SAT16]

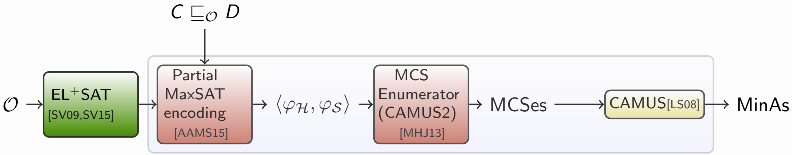
How? - MUS enumeration of Horn formulae (cont.)

- EL2MCS: First effort based on **explicit** hitting set dualization
  - Compute **all** MCSes (MaxSAT-based MCSes enumeration) [MHJ13]
  - Use hitting set dualization to enumerate MUSes [R87,BL03,BS05,LS08]



How? - MUS enumeration of Horn formulae (cont.)

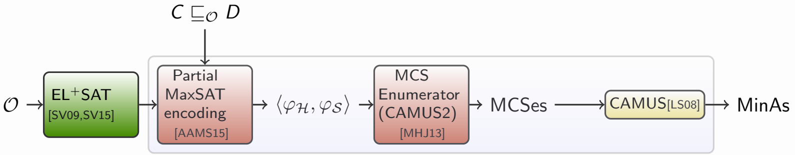
- EL2MCS: First effort based on **explicit** hitting set dualization
  - Compute **all** MCSes (MaxSAT-based MCSes enumeration) [MHJ13]
  - Use hitting set dualization to enumerate MUSes [R87,BL03,BS05,LS08]



- In practice, **EL2MCS extensively outperforms EL$^+$SAT** [AMMS15]
  - But, there can be **exponentially many** MCSes
    - It may even be infeasible to start enumeration of MUSes !

How? - MUS enumeration of Horn formulae (cont.)

- EL2MCS: First effort based on **explicit** hitting set dualization
  - Compute **all** MCSes (MaxSAT-based MCSes enumeration) [MHJ13]
  - Use hitting set dualization to enumerate MUSes [R87,BL03,BS05,LS08]



- In practice, **EL2MCS extensively outperforms $\mathsf{EL}^+\mathsf{SAT}$** [AMMS15]
  - But, there can be **exponentially many** MCSes
    - It may even be infeasible to start enumeration of MUSes **!**
- Alternative is **implicit** hitting set dualization
  - Recently proposed in eMUS / MARCO [PMS13,LM13,LPMMS15]

Performance issues with EL$^+$SAT

- Blocking of MCSes/MSSes does **not** eliminate supersets of MCSes (or subsets of MSSes)
  - **Why?** Clause learning **only** uses decision variables; learned clauses only contain **negative** literals; supersets of MCSes **not** blocked

Performance issues with EL$^+$SAT

- Blocking of MCSes/MSSes does **not** eliminate supersets of MCSes (or subsets of MSSes)
  - **Why?** Clause learning **only** uses decision variables; learned clauses only contain **negative** literals; supersets of MCSes **not** blocked
- Horn formulae decided by unit propagation with CDCL SAT solver
  - In MiniSat, unit propagation takes worst-case quadratic time, due to implementation of watched literals [G13]
  - Also, unnecessary propagation of 0-valued variables [DG84,M88]
  - We can do better: use dedicated LTUR algorithm [M88]

Performance issues with EL$^+$SAT

- Blocking of MCSes/MSSes does **not** eliminate supersets of MCSes (or subsets of MSSes)
  - **Why?** Clause learning **only** uses decision variables; learned clauses only contain **negative** literals; supersets of MCSes **not** blocked
- Horn formulae decided by unit propagation with CDCL SAT solver
  - In MiniSat, unit propagation takes worst-case quadratic time, due to implementation of watched literals [G13]
  - Also, unnecessary propagation of 0-valued variables [DG84,M88]
  - We can do better: use dedicated LTUR algorithm [M88]
- Computing each MinA corresponds to deletion-based MUS extraction
  - We can do better: use **insertion**-based MUS extraction

Performance issues with EL$^+$SAT

- Blocking of MCSes/MSSes does **not** eliminate supersets of MCSes (or subsets of MSSes)
  - **Why?** Clause learning **only** uses decision variables; learned clauses only contain **negative** literals; supersets of MCSes **not** blocked
- Horn formulae decided by unit propagation with CDCL SAT solver
  - In MiniSat, unit propagation takes worst-case quadratic time, due to implementation of watched literals [G13]
  - Also, unnecessary propagation of 0-valued variables [DG84,M88]
  - We can do better: use dedicated LTUR algorithm [M88]
- Computing each MinA corresponds to deletion-based MUS extraction
  - We can do better: use **insertion**-based MUS extraction
- Maximal models computed by assigning decision variables to 1
  - Corresponds to SAT with preferences [GM06,RGM10]
  - We can do better: use MCS extraction [MSHJPB13,MPMS15]

An example with MUS Enumeration of Horn formula

$G_0 = \{(\neg a \vee \neg b), (b), (\neg c \vee \neg d \vee \neg e)\}$
$G_1 = \{(a)\}$
$G_2 = \{(c)\}$
$G_3 = \{(d)\}$
$G_4 = \{(e)\}$

$\text{MUS} = \{\{(a)\}, \{(c), (d), (e)\}\}$
$\text{MCS} = \{\{(a), (c)\}, \{(a), (d)\}, \{(a), (e)\}\}$

- Blocking with negative clauses [VS09,VS15]

Q	MxM model	MUS	MCS	Blocking clause	OK?
$\emptyset$	$p_1 = \ldots = 1$	$\{G_1\}$		$B_1 = (\neg p_1)$	✓
$\{B_1\}$	$p_2 = \ldots = 1$	$\{G_2, G_3, G_4\}$		$B_2 = (\neg p_2 \vee \neg p_3 \vee \neg p_4)$	✓
$\{B_1, B_2\}$	$p_3 = p_4 = 1$		$\{G_1, G_2\}$	$B_3 = (\neg p_3 \vee \neg p_4)$	✓
$\{B_1, B_2, B_3\}$	$p_2 = p_3 = 1$		$\{G_1, G_4\}$	$B_4 = (\neg p_2 \vee \neg p_3)$	✓
$\{B_1, \ldots, B_4\}$	$p_2 = p_4 = 1$		$\{G_1, G_3\}$	$B_5 = (\neg p_2 \vee \neg p_4)$	✓
$\{B_1, \ldots, B_5\}$	$p_3 = 1$		$\{G_1, G_2, G_4\}$	$B_6 = (\neg p_3)$	✗
$\{B_1, \ldots, B_6\}$	$p_4 = 1$		$\{G_1, G_2, G_3\}$	$B_7 = (\neg p_4)$	✗
$\{B_1, \ldots, B_7\}$	$p_1 = \ldots = 0$				

An example with eMUS / MARCO (cont.)

$G_0 = \{(\neg a \vee \neg b), (b), (\neg c \vee \neg d \vee \neg e)\}$
$G_1 = \{(a)\}$
$G_2 = \{(c)\}$
$G_3 = \{(d)\}$
$G_4 = \{(e)\}$

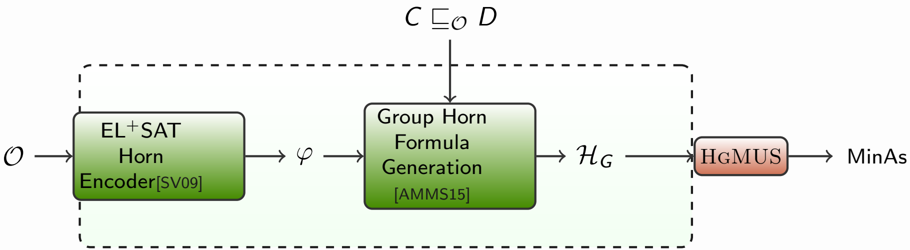
$\text{MUS} = \{\{(a)\}, \{(c), (d), (e)\}\}$
$\text{MCS} = \{\{(a), (c)\}, \{(a), (d)\}, \{(a), (e)\}\}$

- Distinct blocking of MUSes and MCSes [PMS13,LM13,LPMMS13]

Q	MxM model	MUS	MCS	Blocking clause	OK?
$\emptyset$	$p_1 = \ldots = 1$	$\{G_1\}$		$B_1 = (\neg p_1)$	✓
$\{B_1\}$	$p_2 = \ldots = 1$	$\{G_2, G_3, G_4\}$		$B_2 = (\neg p_2 \vee \neg p_3 \vee \neg p_4)$	✓
$\{B_1, B_2\}$	$p_3 = p_4 = 1$		$\{G_1, G_2\}$	$B_3 = (p_1 \vee p_2)$	✓
$\{B_1, B_2, B_3\}$	$p_2 = p_3 = 1$		$\{G_1, G_4\}$	$B_4 = (p_1 \vee p_4)$	✓
$\{B_1, \ldots, B_4\}$	$p_2 = p_4 = 1$		$\{G_1, G_3\}$	$B_5 = (p_1 \vee p_3)$	✓
$\{B_1, \ldots, B_5\}$	$\emptyset$				

Organization of EL2MUS

EL2MUS, front-end to HGMUS, is an efficient tool to solve the problem of axiom pinpointing in Lightweight DLs [AMMS15]



Organization of HgMUS

HgMUS is a state-of-the-art group-MUS enumerator [AMMS15]

Organization of HgMUS

HgMUS is a state-of-the-art group-MUS enumerator [AMMS15]

1. Partial MUS enumeration paradigm [PMS13,LM13,LPMMS15]
   - Implements implicit hitting set dualization

Organization of HGMUS

HGMUS is a state-of-the-art group-MUS enumerator [AMMS15]

1. Partial MUS enumeration paradigm [PMS13,LM13,LPMMS15]
   - Implements implicit hitting set dualization

2. Blocking of MUSes & MCSes

Organization of HgMUS

HgMUS is a state-of-the-art group-MUS enumerator [AMMS15]

1. Partial MUS enumeration paradigm [PMS13,LM13,LPMMS15]
   - Implements implicit hitting set dualization
2. Blocking of MUSes & MCSes
3. Dedicated Horn decision procedure – LTUR [M88]

Organization of HgMUS

HgMUS is a state-of-the-art group-MUS enumerator [AMMS15]

1. Partial MUS enumeration paradigm [PMS13,LM13,LPMMS15]
   - Implements implicit hitting set dualization

2. Blocking of MUSes & MCSes

3. Dedicated Horn decision procedure – LTUR [M88]

4. Optimized enumeration of maximal models
   - Builds on MCS extraction work – MCSls & LBX [MSHJPB13,MPMS15]

Organization of HgMUS

HgMUS is a state-of-the-art group-MUS enumerator [AMMS15]

1. Partial MUS enumeration paradigm [PMS13,LM13,LPMMS15]
   - Implements implicit hitting set dualization

2. Blocking of MUSes & MCSes

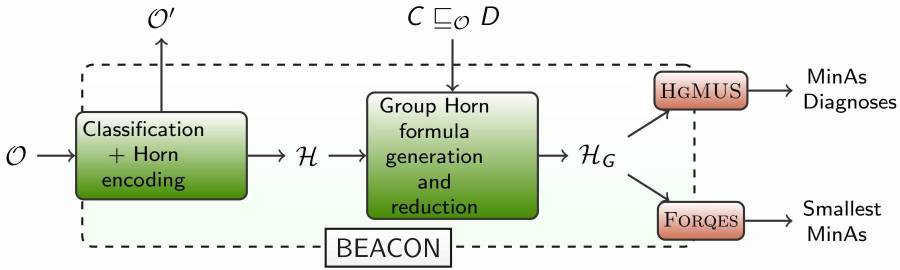
3. Dedicated Horn decision procedure – LTUR [M88]

4. Optimized enumeration of maximal models
   - Builds on MCS extraction work – MCSls & LBX [MSHJPB13,MPMS15]

5. Dedicated MUS extraction algorithm for Horn formulae

BEACON Organization

BEACON is a standalone efficient Debugging tool for $\mathcal{EL}^+$ ontologies that assimilates HGMUS and Forqes [AMMS15,ILPM15, AMMS16]



Contents

What? - Problem

Why? - Relevance

How? - Solution

Results

Conclusions

Publications

Experimental setup

- Medical ontologies:
  - GALEN, with two variants: FULL-GALEN and NOT-GALEN
  - GENE
  - NCI
  - SNOMED CT

Experimental setup

- Medical ontologies:
  - GALEN, with two variants: FULL-GALEN and NOT-GALEN
  - GENE
  - NCI
  - SNOMED CT – **large size!**

Experimental setup

- Medical ontologies:
  - GALEN, with two variants: FULL-GALEN and NOT-GALEN
  - GENE
  - NCI
  - SNOMED CT – **large size!**
- Instance generation for each medical ontology:
  - 50 random
  - 50 sorted – expected to have large number of minimal explanations
  - **Total**: **500** instances

Experimental setup

- Medical ontologies:
  - GALEN, with two variants: FULL-GALEN and NOT-GALEN
  - GENE
  - NCI
  - SNOMED CT – **large size!**
- Instance generation for each medical ontology:
  - 50 random
  - 50 sorted – expected to have large number of minimal explanations
  - **Total**: **500** instances
- Instance simplifications: [VS09,VS15]
  - COI: Cone-of-influence reduction
  - x2: apply COI, re-encode to ontology and encode to Horn formula

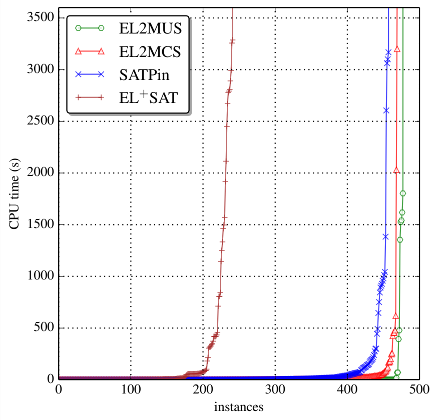
Experimental setup

- Medical ontologies:
  - GALEN, with two variants: FULL-GALEN and NOT-GALEN
  - GENE
  - NCI
  - SNOMED CT – **large size!**
- Instance generation for each medical ontology:
  - 50 random
  - 50 sorted – expected to have large number of minimal explanations
  - **Total**: **500** instances
- Instance simplifications: [VS09,VS15]
  - COI: Cone-of-influence reduction
  - x2: apply COI, re-encode to ontology and encode to Horn formula
- Compute cluster, with 3600s time limit and 4 GByte memory limit

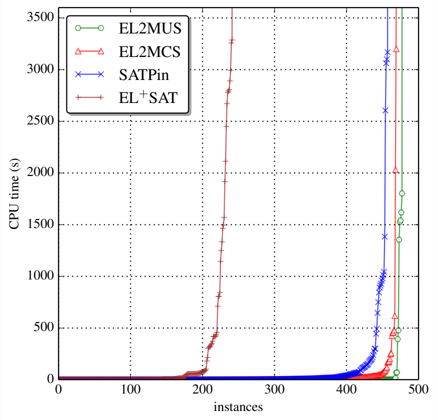
Experimental setup – goal & tools

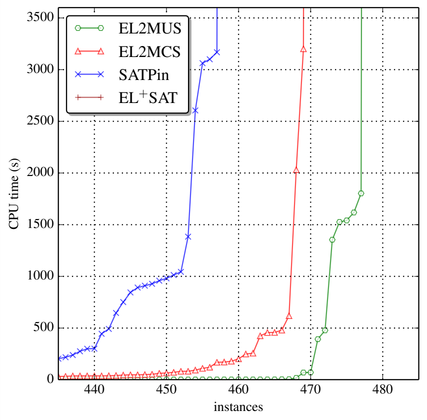
- Goal:
  - Compute **all** MinAs/MUSes or as many MinAs/MUSes as possible within the given timeout
- Our tools:
  - **EL2MCS**: (first attempt) developed in 2015 [AMMS15]
  - **EL2MUS**: uses EL^{+}SAT encoder as front-end to **HgMUS** [AMMS15]
  - **BEACON**: standalone complete tool [AMMS16]
- Other SAT-based tools:
  - EL^{+}SAT: developed in 2009, updated in 2014, 2015 [VS09,VS15]
  - SATPin: developed in 2015 [MP15]
- Other (non SAT-based) tools:
  - CEL – only computes 10 MinAs, developed in 2006 [BLS06]
  - JUST – cannot handle all $\mathcal{EL}^{+}$ constructs, developed in 2014 [L14]

Cactus plots – COI instances

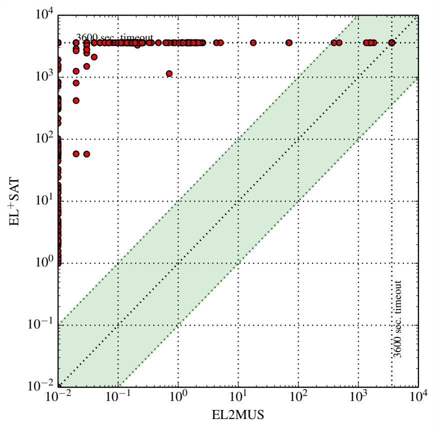


Cactus plots – COI instances

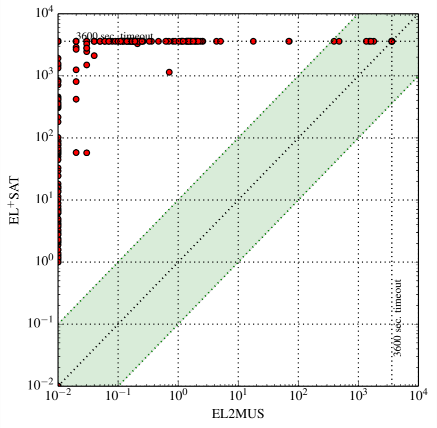


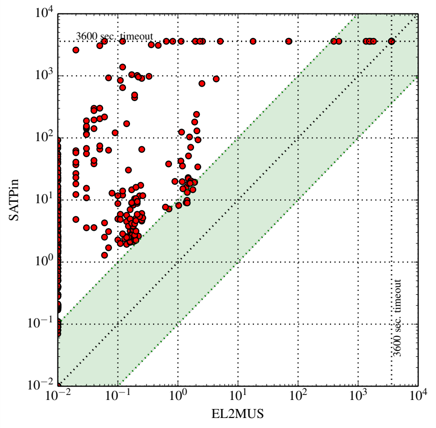


Scatter plots – COI instances

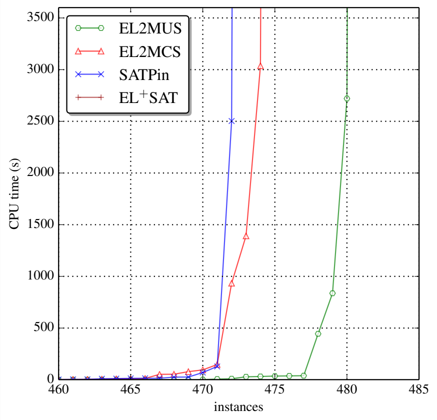


Scatter plots – COI instances

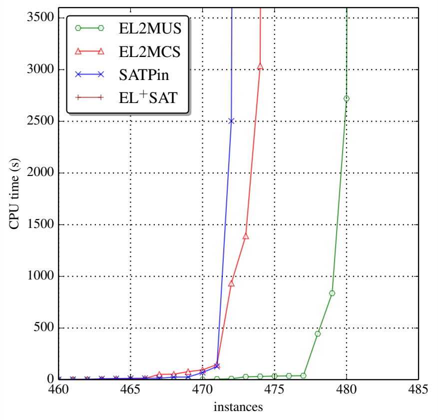


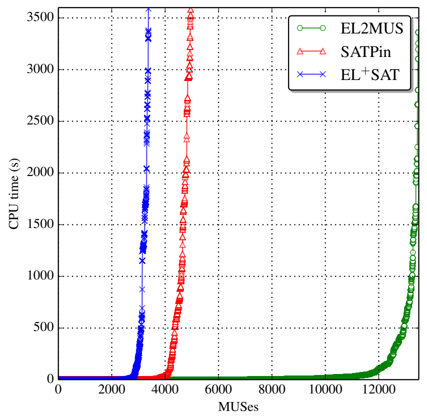


Cactus plots – x2 instances

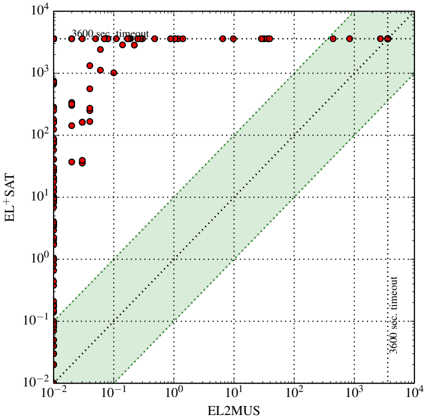


Cactus plots – x2 instances

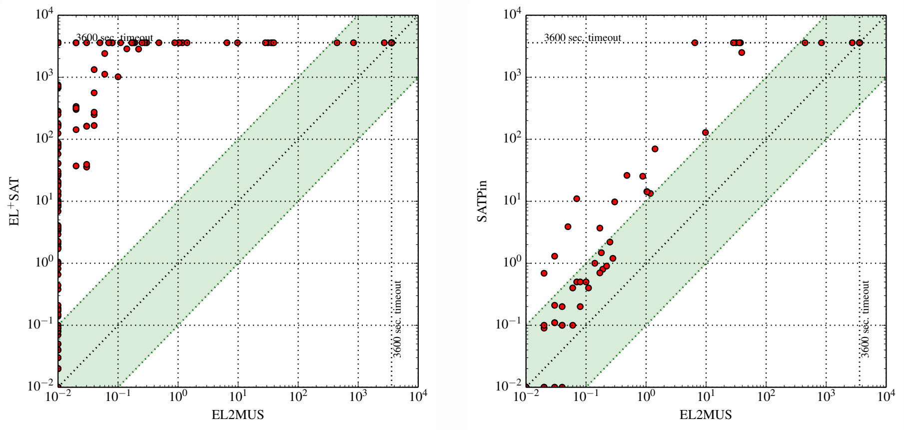




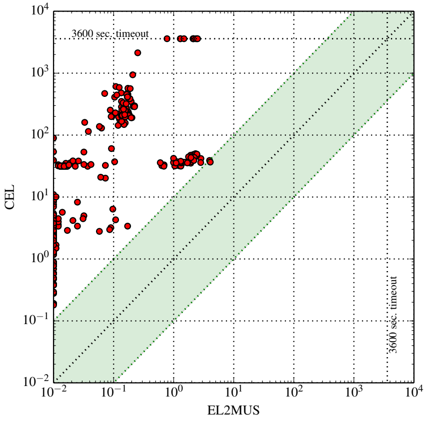
Scatter plots – x2 instances



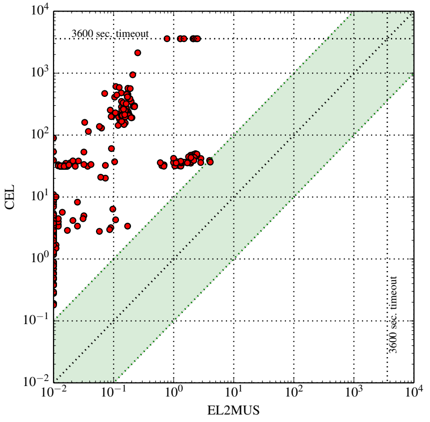
Scatter plots – x2 instances

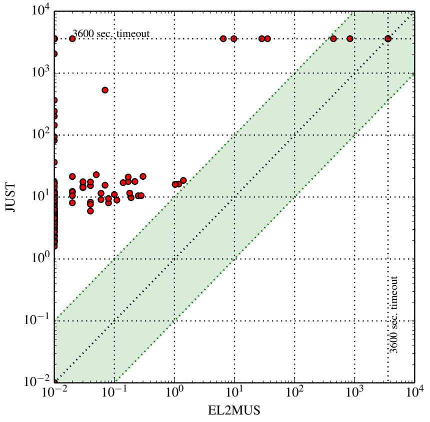


Non-SAT based tools – vs CEL on COI instances



Non-SAT based tools – vs Just on x2 instances





Contents

What? - Problem

Why? - Relevance

How? - Solution

Results

Conclusions

Publications

Conclusions & research directions

- Developed state-of-the art debugging and axiom pinpointing tools
- Novel MUS enumeration algorithm for group Horn formulae
  - Exploits partial MUS enumeration, MCS extraction, (old) LTUR, insertion-based MUS extraction, etc.
- Clear performance gains (Orders of magnitude speedups for the larger instances)
- Can enumerate **much** larger number of MUSes/MinAs than any other approach

- Integrate recent advances in MUS/MCS extraction & enumeration
- Find more applications

Contents

What? - Problem

Why? - Relevance

How? - Solution

Results

Conclusions

Publications

Publications

1. **M. Fareed Arif**, Carlos Mencía, and Joao Marques-Silva. Efficient Axiom Pinpointing with EL2MCS. In KI 2015, volume 9324, pages 225-233, Springer 2015
   - KI 2015: CORE C conference, but with attendance by many DL researchers
2. **M. Fareed Arif**, Carlos Mencía, and Joao Marques-Silva. Efficient MUS Enumeration of Horn Formulae with Applications to Axiom Pinpointing. In SAT 2015, volume 9340, pages 324-342. Springer 2015
   - SAT 2015: CORE A conference, representing the bulk of the scientific contributions.
3. **M. Fareed Arif**, Carlos Mencía, Alexey Ignatiev, Norbert Manthey, Rafael Peñaloza, and Joao Marques-Silva. BEACON: An Efficient SAT-Based Tool for Debugging EL+ Ontologies. In SAT 2016, volume 9710, pages 521-530. Springer 2015
   - SAT 2016: CORE A conference, detailing the final result of the work.
4. **M. Fareed Arif**, Carlos Mencía Norbert Manthey, Alexey Ignatiev, Rafael Peñaloza, and Joao Marques-Silva. Towards Efficient SAT-Based Axiom Pinpointing in Lightweight Description Logics. JAIR 2016 (under review)
   - JAIR 2016: submitted for an A* journal paper, currently under review.

Thank You