Intelligent Tableaux Algorithm for Descrition Logic Reasoning

Ming Zuo Volker Haarslev

Concordia University, Montreal

September, 2013

Outline

- Motivation
- Related Work
- $\mathcal{ALC}$ DL Syntax and Semantics
- Normalization
- Local Learning and Global Learning
- Forgetting
- Experimental Results
- Future Work

Motivation

- Tremendous amount of theoretical and Implementation work
  - Tableaux, Automata, SMT, Resolution $\cdots$
  - FACT++, Pellet, RACER, HermiT $\cdots$
  - Comprehensive sets of optimization algorithms and techniques
- Efficient reasoning is the gap
  - Two sources of complexity: OR-Branching, AND-Branching
  - An active area of research
- $\mathcal{ALC}$ is the expressive DL with the simplest syntax which well addresses the two sources of complexity

Related Work

- Tableaux Based Algorithms
  - De-facto standard algorithm
  - Simple problems optimal and difficult problems suboptimal
- Tableaux with global cache
  - Simple problem suboptimal and difficult problem optimal
- Automata based algorithm
  - EXPTime, EXPSpace
- SMT based solution
  - Early stage in DL

$\mathcal{ALC}$ DL Syntax and Semantics

- Attributive Concept Language with Complements
  - Signature $\Sigma = (\mathcal{N}_R, \mathcal{N}_C, \mathcal{N}_I)$
  - Concepts $C, D ::= \top \mid \bot \mid A \mid \neg C \mid C \sqcap D \mid C \sqcup D \mid \forall R.C \mid \exists R.C$
- Key Definitions
  - Atomic concept, Atomic literal, Modal Atomic Concept
  - Axiom($C \sqsubseteq D, C \equiv D$) and TBox denoted as $\mathcal{T}$
  - Assertion $(a : C, R(a, b))$ and ABox denoted as $\mathcal{A}$
  - A Knowledge Base(KB) is a tuple $\mathcal{K} = \{\mathcal{A}, \mathcal{T}\}$
- Interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ where $\Delta^{\mathcal{I}}$ is a non-empty set and $\cdot^{\mathcal{I}}$ is a mapping $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}; C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}; d^{\mathcal{I}} \in \Delta^{\mathcal{I}}$.
- An interpretation $\mathcal{I}$ is called a model of $\mathcal{K}$ if it satisfies every assertions in $\mathcal{A}$ and validates every axiom in $\mathcal{T}$.
- $\mathcal{K}$ is satisfiable if it has at least one model.
- All DL reasoning problems P-reducable to $\mathcal{K}$ SAT problem.

Normalization

- Description Logic Normal Form (DLNF)
  - $d : \bigsqcup_1^i A_i$
  - $A \sqsubseteq \bigsqcup_1^i B_i$
  - $A_1 \sqsubseteq \exists R.B$
  - $A_2 \sqsubseteq \forall R.B$

    where d is an individual and A_i, B_i are atomic literals. A_1, A_2 are called prefixes.

- Proposition

  For every KB $\mathcal{K}$, one can compute in linear time a KB $\mathcal{K}'$ in DLNF such that $\mathcal{K}$ is satisfiable iff $\mathcal{K}'$ is satisfiable (equi-satisfiable)

Normalization (Continued)

- De Morgan's Laws
  - $\cdots$
- Simplification Rules
  - $C \sqcap C = C$
  - $\neg\neg C = C \cdots$
- Normalization Rules
  - $a : C \Rightarrow \{a : X, X \sqsubseteq C\}$ where $X \notin \mathcal{N}_C$ of $\mathcal{K}'$
  - $A \sqsubseteq C \sqcup D \Rightarrow \{A \sqsubseteq X \sqcup Y, X \sqsubseteq C, Y \sqsubseteq D\}$ where $X, Y \notin \mathcal{N}_C$ of $\mathcal{K}'$
  - $A \sqsubseteq \exists R.C \Rightarrow \{A \sqsubseteq \exists R.X, X \sqsubseteq C\}$ where $X \notin \mathcal{N}_C$ of $\mathcal{K}'$
  - $A \sqsubseteq \forall R.C \Rightarrow \{A \sqsubseteq \forall R.X, X \sqsubseteq C\}$ where $X \notin \mathcal{N}_C$ of $\mathcal{K}'$
  - $A \sqsubseteq C \sqcap D \Rightarrow \{A \sqsubseteq C, A \sqsubseteq D\}$

Local Learning

- OR-branching in DL (Tableau $\sqcup$-rule/β-rule):
  $\frac{a:(C \sqcup D) \in \mathcal{K}}{\mathcal{K} \cup (a:C) \mid \mathcal{K} \cup (a:D)}$
  - Mathematically correct
  - Workable for general KB
  - Practically inefficient
    - $\mathcal{K}$ is copied to every branch
    - Worst case up to $O(n^{2^n})$ while the $\sqcup$ is NP per se.
- Solution in the literature
  - Open research topic
  - DPLL with conflict-driven learning with high-performance in reality $O(2^n)$
  - No state-of-the-art DL reasoners deal with $\sqcup$ efficiently so far

Local Learning (Continued)

- Our solution
  - Group all assertions in a DLNF $\mathcal{K}$ by individual (Node)
  - Satisfiability of each node becomes SAT w.r.t. a CNF
  - DPLL on CNF with conflict-driven learning (local learning), two-watched-literals etc.
  - A node model in the form of $d : \{B_1, B_2 .... B_i\}$ where B_i is an atomic literal
- Local learning is out of the scope in DL research

Example

$A, B, C, D, E, X \in \mathcal{N}_C$ and $R \in \mathcal{N}_R$ of $\mathcal{K}$

1. $a : X$
2. $B \sqsubseteq A$
3. $C \sqsubseteq A \sqcup B$
4. $A \sqsubseteq D$
5. $A \sqsubseteq E$
6. $D \sqsubseteq \exists R.X$
7. $E \sqsubseteq \forall R.\neg X$
8. $\top \sqsubseteq A \sqcup B \sqcup C$

Proof:

Node	*Model*
$a : \begin{pmatrix} X \\ A \sqcup B \sqcup C \end{pmatrix}$	$\begin{pmatrix} 0 : X \end{pmatrix}$
$a : \begin{pmatrix} X \\ A \sqcup B \sqcup C \\ D, E \\ \forall R.\neg X, \exists R.X \end{pmatrix}$	$\begin{pmatrix} 0 : X \\ 1 : A, D, E, \\ \forall R.\neg X, \exists R.X \\ R(a, b) \end{pmatrix}$

Example(Continued)

Node $\qquad$ *Model*

$$b : \begin{pmatrix} X \\ \neg X \\ A \sqcup B \sqcup C \end{pmatrix} \quad \begin{pmatrix} 0 : \end{pmatrix}$$

$$\mathcal{K} = \mathcal{K} \bigcup \{\top \sqsubseteq \neg D \sqcup \neg E\}$$

Node $\qquad$ *Model*

$$a : \begin{pmatrix} X \\ A \sqcup B \sqcup C \\ \neg D \sqcup \neg E \end{pmatrix} \quad \begin{pmatrix} 0 : X, \neg A \end{pmatrix}$$

Node $\qquad$ *Model*

$$a : \begin{pmatrix} X \\ A \sqcup B \sqcup C \\ \neg D \sqcup \neg E \\ D, E \\ \forall R.\neg X, \exists R.X \end{pmatrix} \quad \begin{pmatrix} 0 : X \\ 1 : A, D, E, \\ \forall R.\neg X, \exists R.X \end{pmatrix} \quad \ldots\ldots$$

Global Learning

- Restrictions of the node model:
  1. $\frac{a:B\in\mathcal{M}_a,(B\sqsubseteq C)\in\mathcal{T}}{\mathcal{A}\cup\{a:C\}}$, $\frac{a:B\in\mathcal{M}_a,(\top\sqsubseteq C)\in\mathcal{T}}{\mathcal{A}\cup\{a:C\}}$ (unfolding-rule)
  2. $\frac{a:\exists R.C\in\mathcal{M}_a}{\Delta\cup\{b\},\mathcal{A}\cup\{R(a,b),b:C\}}$ ($\exists$-rule)

     a is a parent node of b; C is an label of b; R is an edge from a to b.
  3. $\frac{a:\forall R.C\in\mathcal{M}_a}{R(a,b)\in\mathcal{A}\rightarrow\mathcal{A}\cup\{b:C\}}$ ($\forall$-rule)

     C is a universal label of node b; empty label node is called root.

Global Learning (Continued)

- Interesting facts:
  - All nodes have to be satisifiable at the same time if $\mathcal{K}$ is satisfiable (AND-node).
  - Due to unfolding-rule, node content is dynamic at run-time, but satisfiability is determined by its label set.
  - Label set is uniquely determined by its prefix set which is a subset of its parent's node model.
- Globle learning
  - UNSAT-learning
  - Unknown-learning (remembering)
  - SAT-learning (remembering)

Forgetting

- Forgetting proposition. Let $\mathcal{K}$ a KB and ϕ a formula, and $\mathcal{K}' = \mathcal{K} \setminus \{\phi\}$. If $\mathcal{K}' \models \phi$, then $\mathcal{K}$ and $\mathcal{K}'$ are equisatisfiable.
- Forgetting ensures practical tractability.
- Methods of forgetting
  - FIFO
  - Heuristic
  - Advanced algorithms

Experimental Results[1]

	L-N	L-S	L-U	LIGHT	HermiT	Pellet	Fact++	Racer
galen2s	0.16	0.15	0.16	0.15	1.3	1.4	0.46	1.9
JNH16u	0.07	0.06	0.06	0.07	237.4	452.3	MO	15384
k_d4_13nu	TO	TO	99.77	98.70	TO	TO	TO	TO
k_dum_19nu	TO	TO	37.59	32.27	TO	TO	MO	140.88
k_ph_14pu	963.7	1001	1005	1014	MO	MO	MO	TO
k_tp4_21nu	15.84	15.31	5.32	0.32	TO	0.54	MO	TO
k_branch_21nu	0.39	0.40	0.40	0.39	TO	2.4	18.2	19.2
k_path_21pu	1.7	1.76	0.23	1.78	MO	25.63	7.30	9.0
k_poly_16pu	MO	MO	MO	0.61	373.4	76.98	MO	2.03
k_poly_21ns	MO	MO	TO	325.4	MO	MO	MO	524.6
BCS4s	TO	1.37	TO	0.20	133.8	TO	MO	13.8
BCS5s	TO	TO	TO	2.14	MO	TO	MO	276.2

[1] All numbers are in second. TO—Time Out; MO—Memory Out

Conclusion and Future Work

- Conclusion
  - DPLL with local learning is efficient w.r.t. a single node
  - Global Learning is effective compared to any existing algorithms
  - Optimization on difficult problems has no obvious impact to simple problems
  - Compatible with lazy unfolding
- Future Work
  - Integration with existing optimization algorithms/techniques
  - Optimization on learning and forgetting
  - Application to more expressive DL

Q & A

- Questions?