

# Locality and Subsumption Testing in $\mathcal{EL}$ and some of its extensions

Viorica Sofronie-Stokkermans

Max-Planck-Institut für Informatik

Saarbrücken

AiML 2008, September 9–12, 2008, Nancy

# Motivation

---

Description logics – used for KR in databases/ontologies

Provide a logical basis for modeling and reasoning about:

- objects
- classes of objects (**concepts**)
- relationships between objects (**links, roles**)

Wide variety of description logics (various degrees of expressivity)

**This talk:** Tractable description logic:  $\mathcal{EL}$ ,  $\mathcal{EL}^+$  and extensions  
[Baader'03–] used e.g. in medical ontologies (SNOMED)

# Motivation

---

Description logics – used for KR in databases/ontologies

**Focus:** Tractable description logic:  $\mathcal{EL}$ ,  $\mathcal{EL}^+$  and extensions [Baader'03–]  
used e.g. in medical ontologies (SNOMED)

## Main contributions of this paper:

- Alternative proof of tractability of  $\mathcal{EL}$ ,  $\mathcal{EL}^+$  based on a notion of locality
  - ↳ a hierarchical reduction to SAT in the case of  $\mathcal{EL}$
  - ↳ reduction to SAT of ground Horn formulae in the case of  $\mathcal{EL}^+$
- Identify tractable extensions of  $\mathcal{EL}$  and  $\mathcal{EL}^+$ 
  - ↳ with  $n$ -ary roles and/or numerical domains.

# Overview

---

- **Introduce  $\mathcal{EL}$  and  $\mathcal{EL}^+$  and the deduction problems**  
(subsumption w.r.t. a TBox resp. CBox)
- **Algebraic semantics**  
TBox/CBox subsumption  $\mapsto$  u.w.p. for  $\text{SLat} \cup \text{Mon}(\Sigma)(\cup Ax)$
- **Local theories and local theory extensions**
- **Locality results for  $\text{SLat} \cup \text{Mon}(\Sigma)(\cup Ax)$**
- **Tractable extensions of  $\mathcal{EL}, \mathcal{EL}^+$**

# Overview

---

- **Introduce  $\mathcal{EL}$  and  $\mathcal{EL}^+$  and the deduction problems**  
(subsumption w.r.t. a TBox resp. CBox)
- **Algebraic semantics**  
TBox/CBox subsumption  $\mapsto$  u.w.p. for  $\text{SLat} \cup \text{Mon}(\Sigma)(\cup \text{Ax})$
- **Local theories and local theory extensions**
- **Locality results for  $\text{SLat} \cup \text{Mon}(\Sigma)(\cup \text{Ax})$**
- **Tractable extensions of  $\mathcal{EL}, \mathcal{EL}^+$**

# $\mathcal{EL}$ : Generalities

---

- Concepts:**
- primitive concepts  $N_C$
  - complex concepts (built using concept constructors  $\sqcap, \exists r$ )

**Roles:**  $N_R$

- Interpretations:**  $\mathcal{I} = (D^{\mathcal{I}}, \cdot^{\mathcal{I}})$
- $C \in N_C \mapsto C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
  - $r \in N_R \mapsto r^{\mathcal{I}} \subseteq D^{\mathcal{I}} \times D^{\mathcal{I}}$

Constructor name	Syntax	Semantics
conjunction	$C_1 \sqcap C_2$	$C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}}$
existential restriction	$\exists r.C$	$\{x \mid \exists y((x, y) \in r^{\mathcal{I}} \text{ and } y \in C^{\mathcal{I}})\}$

# $\mathcal{EL}$ : Generalities

---

- Concepts:**
- primitive concepts  $N_C$
  - complex concepts (built using concept constructors  $\sqcap, \exists r$ )

**Roles:**  $N_R$

- Interpretations:**  $\mathcal{I} = (D^{\mathcal{I}}, \cdot^{\mathcal{I}})$
- $C \in N_C \mapsto C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
  - $r \in N_R \mapsto r^{\mathcal{I}} \subseteq D^{\mathcal{I}} \times D^{\mathcal{I}}$

## **Problem:**

**Given:** TBox (set  $\mathcal{T}$  of concept inclusions  $C_i \sqsubseteq D_i$ ) and concepts  $C, D$

**Task:** test whether  $C \sqsubseteq_{\mathcal{T}} D$ ,

i.e. whether for all  $\mathcal{I}$  **if**  $C_i^{\mathcal{I}} \subseteq D_i^{\mathcal{I}}$  **then**  $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$   
 $\forall C_i \sqsubseteq D_i \in \mathcal{T}$

Decidable in PTIME [Baader02] (graphs/simulation, transl. to Datalog, ...)

# $\mathcal{EL}$ : Example

---

Primitive concepts:	protein, process, substance
Roles:	catalyzes, produces
Terminology: (TBox)	enzyme = protein $\sqcap$ $\exists$ catalyzes.reaction catalyzer = $\exists$ catalyzes.process reaction = process $\sqcap$ $\exists$ produces.substance
Query:	enzyme $\sqsubseteq$ catalyzer?

# Algebraic semantics for $\mathcal{EL}$

Translation of concept descriptions to terms:

$$\begin{aligned} C &\mapsto \bar{C} \text{ for any concept name } C \\ C_1 \sqcap C_2 &\mapsto \overline{C_1 \sqcap C_2} = \bar{C}_1 \wedge \bar{C}_2 \\ \exists r.C &\mapsto \overline{\exists r.C} = f_r(\bar{C}) \end{aligned}$$

## Algebraic semantics for $\mathcal{EL}$ .

Assume that  $N_C = \{c_1, \dots, c_n\}$ . The following are equivalent:

- (1)  $C \sqsubseteq_{\mathcal{T}} D$
- (2)  $\text{BAO} \cup \text{Jh}(\{f_r \mid r \in N_R\}) \models \forall c_1, \dots, c_n ((\bigwedge_{C_i \sqsubseteq D_i \in \mathcal{T}} \bar{C}_i \leq \bar{D}_i) \rightarrow \bar{C} \leq \bar{D})$
- (3)  $\text{SLat} \cup \text{Mon}(\{f_r \mid r \in N_R\}) \models \forall c_1, \dots, c_n ((\bigwedge_{C_i \sqsubseteq D_i \in \mathcal{T}} \bar{C}_i \leq \bar{D}_i) \rightarrow \bar{C} \leq \bar{D})$

# Algebraic semantics for $\mathcal{EL}$

Translation of concept descriptions to terms:

$$\begin{aligned} C &\mapsto \bar{C} \text{ for any concept name } C \\ C_1 \sqcap C_2 &\mapsto \overline{C_1 \sqcap C_2} = \bar{C}_1 \wedge \bar{C}_2 \\ \exists r.C &\mapsto \overline{\exists r.C} = f_r(\bar{C}) \end{aligned}$$

## Algebraic semantics for $\mathcal{EL}$ .

Assume that  $N_C = \{c_1, \dots, c_n\}$ . The following are equivalent:

- (1)  $C \sqsubseteq_{\mathcal{T}} D$
- (2)  $\text{BAO} \cup \text{Jh}(\{f_r \mid r \in N_R\}) \models \forall c_1, \dots, c_n ((\bigwedge_{C_i \sqsubseteq D_i \in \mathcal{T}} \bar{C}_i \leq \bar{D}_i) \rightarrow \bar{C} \leq \bar{D})$
- (3)  $\text{SLat} \cup \text{Mon}(\{f_r \mid r \in N_R\}) \models \forall c_1, \dots, c_n ((\bigwedge_{C_i \sqsubseteq D_i \in \mathcal{T}} \bar{C}_i \leq \bar{D}_i) \rightarrow \bar{C} \leq \bar{D})$

# $\mathcal{EL}$ : Algebraic semantics

Primitive concepts:	protein, process, substance
Roles:	catalyzes, produces
Terminology: (TBox)	$\text{enzyme} = \text{protein} \sqcap \exists \text{catalyzes}.\text{reaction}$ $\text{catalyzer} = \exists \text{catalyzes}.\text{process}$ $\text{reaction} = \text{process} \sqcap \exists \text{produces}.\text{substance}$
Query:	$\text{enzyme} \sqsubseteq \text{catalyzer}?$

$\text{SLat} \cup \text{Mon} \models \text{enzyme} = \text{protein} \sqcap \text{catalyzes-some}(\text{reaction}) \quad \wedge$   
 $\text{catalyzer} = \text{catalyze-some}(\text{process}) \quad \wedge$   
 $\text{reaction} = \text{process} \sqcap \text{produces-some}(\text{substance})$   
 $\Rightarrow \text{enzyme} \sqsubseteq \text{catalyzer}$

$\text{Mon} : \forall C, D (C \sqsubseteq D \rightarrow \text{catalyze-some}(C) \sqsubseteq \text{catalyze-some}(D))$   
 $\forall C, D (C \sqsubseteq D \rightarrow \text{produces-some}(C) \sqsubseteq \text{produces-some}(D))$

# $\mathcal{EL}^+$ : generalities

---

- Concepts:**
- primitive concepts  $N_C$
  - complex concepts (built using concept constructors  $\sqcap, \exists r$ )

**Roles:**  $N_R$

- Interpretations:**  $\mathcal{I} = (D^{\mathcal{I}}, \cdot^{\mathcal{I}})$
- $C \in N_C \mapsto C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
  - $r \in N_R \mapsto r^{\mathcal{I}} \subseteq D^{\mathcal{I}} \times D^{\mathcal{I}}$

## **Problem:**

**Given:** CBox  $\mathcal{C} = (\mathcal{T}, RI)$ , where  $\mathcal{T}$  set of concept inclusions  $C_i \sqsubseteq D_i$ ;  
 $RI$  set of role inclusions  $r \circ s \sqsubseteq t$  or  $r \sqsubseteq t$

concepts  $C, D$

**Task:** test whether  $C \sqsubseteq_{\mathcal{T}} D$ ,

i.e. whether for all  $\mathcal{I}$  if  $C_i^{\mathcal{I}} \subseteq D_i^{\mathcal{I}}$  and  $r^{\mathcal{I}} \circ s^{\mathcal{I}} \subseteq t^{\mathcal{I}}$  then  $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$   
 $\forall C_i \sqsubseteq D_i \in \mathcal{T} \quad \forall r \circ s \sqsubseteq t \in RI$

# $\mathcal{EL}^+$ : Example

---

Primitive concepts:	protein, process, substance
Roles:	catalyzes, produces, helps-producing
Terminology: (TBox)	enzyme = protein $\sqcap$ $\exists$ catalyzes.reaction reaction = process $\sqcap$ $\exists$ produces.substance
Role inclusions:	catalyzes $\circ$ produces $\sqsubseteq$ helps-producing
Query:	enzyme $\sqsubseteq$ protein $\sqcap$ $\exists$ helps-producing.substance ?

# Algebraic semantics for $\mathcal{EL}^+$

Translation of concept descriptions to terms:

$$C \quad \mapsto \quad \overline{C} \text{ for any concept name } C$$

$$C_1 \sqcap C_2 \quad \mapsto \quad \overline{C_1 \sqcap C_2} = \overline{C_1} \wedge \overline{C_2}$$

$$\exists r.C \quad \mapsto \quad \overline{\exists r.C} = f_r(\overline{C})$$

$$RI \quad \mapsto \quad Ax(RI) = \{\forall x f_r(f_s(x)) \leq f_t(x) \mid r \circ s \subseteq t \in RI\}$$

## Algebraic semantics for $\mathcal{EL}^+$

Assume that  $N_C = \{c_1, \dots, c_n\}$ . T.f.a.e. for any  $\mathcal{EL}^+$  C Box  $\mathcal{C} = (\mathcal{T}, RI)$

$$(1) \quad C \sqsubseteq_{\mathcal{C}} D$$

$$(2) \quad \text{SLatUMon}(\{f_r \mid r \in N_R\}) \cup Ax(RI) \models \forall c_1, \dots, c_n \left( \left( \bigwedge_{C_i \sqsubseteq_{\mathcal{C}} D_i \in \mathcal{T}} \overline{C_i} \leq \overline{D_i} \right) \rightarrow \overline{C} \leq \overline{D} \right)$$

# $\mathcal{EL}^+$ : Algebraic semantics

Primitive concepts:	protein, process, substance
Roles:	catalyzes, produces, helps-producing
Terminology: (TBox)	enzyme = protein $\sqcap$ $\exists$ catalyzes.reaction reaction = process $\sqcap$ $\exists$ produces.substance
Role inclusions:	catalyzes $\circ$ produces $\sqsubseteq$ helps-producing
Query:	enzyme $\sqsubseteq$ protein $\sqcap$ $\exists$ helps-producing.substance ?

$\text{SLat} \cup \text{Mon} \cup \text{Ax}(RI) \models \text{enzyme} = \text{protein} \sqcap \text{catalyzes-some}(\text{reaction}) \wedge$   
 $\text{reaction} = \text{process} \sqcap \text{produces-some}(\text{substance})$   
 $\Rightarrow \text{enzyme} \sqsubseteq \text{protein} \sqcap \text{helps-producing-some}(\text{substance})$

$\text{Mon} : \forall C, D (C \sqsubseteq D \rightarrow \text{catalyze-some}(C) \sqsubseteq \text{catalyze-some}(D))$   
 $\forall C, D (C \sqsubseteq D \rightarrow \text{produces-some}(C) \sqsubseteq \text{produces-some}(D))$

$\text{Ax}(RI) : \forall x (\text{catalyzes-some}(\text{produces-some}(x)) \leq \text{helps-producing}(x))$

# Efficient reasoning in the algebraic models

---

- TBox subsumption in  $\mathcal{EL}$   $\mapsto$  uniform word problem w.r.t.  $SLat \cup Mon(\Sigma)$
- CBox subsumption in  $\mathcal{EL}^+$   $\mapsto$  uniform word problem  $SLat \cup Mon(\Sigma) \cup Ax$ ,

where  $Ax$  consists of axioms of the type:

$$\forall x(f(x) \leq g(x)) \quad \text{and} \quad \forall x(f(g(x)) \leq h(x))$$

**The uniform word problem for  $SLat \cup Mon(\Sigma) \cup Ax$ : decidable in PTIME**

**Explanation:** Local theories / local theory extensions

# Overview

---

- **Introduce  $\mathcal{EL}$  and  $\mathcal{EL}^+$  and the deduction problems**  
(subsumption w.r.t. a TBox resp. CBox)
- **Algebraic semantics**  
TBox/CBox subsumption  $\mapsto$  u.w.p. for  $\text{SLat} \cup \text{Mon}(\Sigma)(\cup Ax)$
- **Local theories and local theory extensions**
- **Locality results for  $\text{SLat} \cup \text{Mon}(\Sigma)(\cup Ax)$**
- **Tractable extensions of  $\mathcal{EL}, \mathcal{EL}^+$**

# Local theories

---

Local theories [McAllester & Givan'92; Ganzinger'01]

$\mathcal{K}$  set of equational Horn clauses

$\mathcal{K}$  is **local**, if for ground clauses  $G$ ,  
 $\mathcal{K} \cup G \models \perp$  iff  $\mathcal{K}[G] \cup G \models \perp$

[McAllester, Givan '92, '93] Local theories capture PTIME

# Local theories

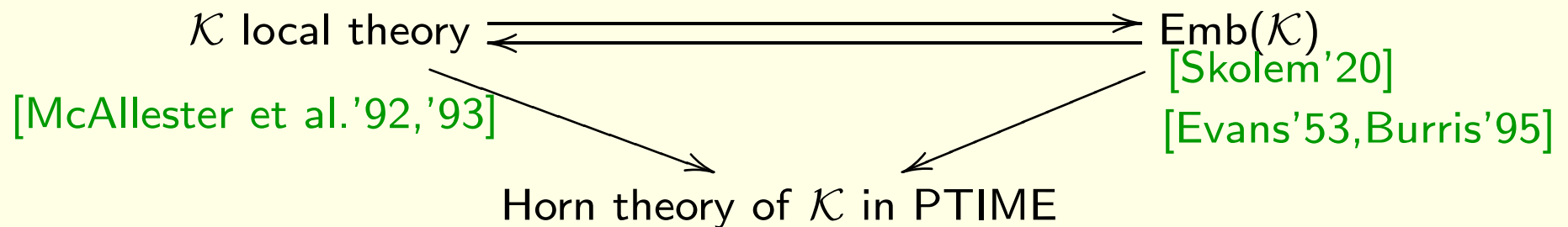
Local theories

[McAllester & Givan'92; Ganzinger'01]

$\mathcal{K}$  set of equational Horn clauses

$\mathcal{K}$  is **local**, if for ground clauses  $G$ ,  
 $\mathcal{K} \cup G \models \perp$  iff  $\mathcal{K}[G] \cup G \models \perp$

[Ganzinger'01]



# Local theories

---

Local theories [McAllester & Givan'92; Ganzinger'01]

$\mathcal{K}$  set of equational Horn clauses

$\mathcal{K}$  is **local**, if for ground clauses  $G$ ,  
 $\mathcal{K} \cup G \models \perp$  iff  $\mathcal{K}[G] \cup G \models \perp$

## Example:

- Axiomatization of lattices [Skolem 1920]

Locality: every poset embeds into a lattice (Dedekind-MacNeille completion)

- Similar results for axiomatizations of semilattices
- Several other examples (algebra [Burris95]; verification: theories of lists ...)

# Local theories

Local theories

[McAllester & Givan'92; Ganzinger'01]

$\mathcal{K}$  set of equational **Horn** clauses

$\mathcal{K}$ is <b>local</b> , if for ground clauses $G$ , $\mathcal{K} \cup G \models \perp$ iff $\mathcal{K}[G] \cup G \models \perp$	<b>Compl:</b> Pol. in size of $G$
	<b>Ex:</b> $G : f(c)=f(d)$ $\text{Mon}_f[G] : c \leq d \rightarrow f(c) \leq f(d)$

$\Psi$  closure operation on ground terms.

$\mathcal{K}$ is <b><math>\Psi</math>-local</b> , if for ground clauses $G$ , $\mathcal{K} \cup G \models \perp$ iff $\mathcal{K}[\Psi(G)] \cup G \models \perp$	<b>Compl:</b> Pol. in size of $\Psi(G)$
---	---

$\mathcal{T}_0 \subseteq \mathcal{K}$ is <b>stably local</b> , if for ground clauses $G$ , $\mathcal{T}_0 \cup \mathcal{K} \cup G \models \perp$ iff $\mathcal{T}_0 \cup \mathcal{K}^{[G]} \cup G \models \perp$	<b>Compl:</b> Pol. in size of $G$
	<b>Ex:</b> $G : f(c)=f(d)$ $\text{Mon}_f^{[G]} : c \leq d \rightarrow f(c) \leq f(d)$ $f(c) \leq d \rightarrow f(f(c)) \leq f(d)$

# Local theory extensions

---

Local theory extensions [Ganzinger, VS, Waldmann'04, VS'05]

$\mathcal{K}$  set of equational clauses;  $\mathcal{T}_0$  theory;  $\mathcal{T}_1 = \mathcal{T}_0 \cup \mathcal{K}$

$\mathcal{T}_0 \subseteq \mathcal{K}$  is **local**, if for ground clauses  $G$ ,  
 $\mathcal{T}_0 \cup \mathcal{K} \cup G \models \perp$  iff  $\mathcal{T}_0 \cup \mathcal{K}[G] \cup G \models \perp$

$\Psi$  closure operation on ground terms. [Ihlemann, Jacobs. VS'08]

$\mathcal{T}_0 \subseteq \mathcal{K}$  is  **$\Psi$ -local**, if for ground clauses  $G$ ,  
 $\mathcal{T}_0 \cup \mathcal{K} \cup G \models \perp$  iff  $\mathcal{T}_0 \cup \mathcal{K}[\Psi(G)] \cup G \models \perp$

Stably local theory extensions are defined similarly.

# Reasoning in local theory extensions

---

**Locality:**  $\mathcal{T}_0 \cup \mathcal{K} \cup G \models \perp$  iff  $\mathcal{T}_0 \cup \mathcal{K}[\Psi(G)] \cup G \models \perp$

**Problem:** Decide whether  $\mathcal{T}_0 \cup \mathcal{K}[\Psi(G)] \cup G \models \perp$

**Solutions:**

**1:** Use  $SMT(\mathcal{T}_0 + UIF)$ : completeness guaranteed only if  $\mathcal{K}[\Psi(G)]$  ground

**2:** Hierarchical reasoning [VS'05]

reduce to satisfiability in  $\mathcal{T}_0$ : applicable in general; sound and complete

↳ parameterized complexity

# Locality and $\mathcal{EL}, \mathcal{EL}^+$

---

**Problem:** Determine if a theory (or a theory extension) is local

**Solution:** Embeddability of partial models into total models

$$\mathcal{T}_1 \text{ } \Psi\text{-local extension of } \mathcal{T}_0 \rightleftarrows \text{Emb}(\mathcal{T}_0, \mathcal{T}_1)$$

extends [Ganzinger'01, GSW'04, VS'05, VS,Ihlemann'07, Ihlemann,Jacobs,VS'08]

# Locality and $\mathcal{EL}, \mathcal{EL}^+$

---

**Problem:** Determine if a theory (or a theory extension) is local

**Solution:** Embeddability of partial models into total models

$$\mathcal{T}_1 \text{ } \Psi\text{-local extension of } \mathcal{T}_0 \begin{array}{c} \longleftarrow \\ \longrightarrow \end{array} \text{Emb}(\mathcal{T}_0, \mathcal{T}_1)$$

extends [Ganzinger'01, GSW'04, VS'05, VS,Ihlemann'07, Ihlemann,Jacobs,VS'08]

This criterion provides a large number of examples:

- **Mathematics:** ext. with monotone functions; Lipschitz functions, ...
- **Verification:** Theories of pointer structures, lists, records, arrays.
- **Kryptography:** Encode/decode with given key

# Locality and $\mathcal{EL}, \mathcal{EL}^+$

---

**Problem:** Determine if a theory (or a theory extension) is local

**Solution:** Embeddability of **partial models** into total models

$$\mathcal{T}_1 \text{ } \Psi\text{-local extension of } \mathcal{T}_0 \rightleftarrows \text{Emb}(\mathcal{T}_0, \mathcal{T}_1)$$

extends [Ganzinger'01, GSW'04, VS'05, VS,Ihlemann'07, Ihlemann,Jacobs, VS'08]

- $P$  is a **weak partial model** of a Horn clause  $C = \bigwedge L_i \rightarrow L$  if for each valuation  $\beta$ , if all terms in  $C$  are defined then  $C$  is true in  $P$ .
- $P$  is an **Evans partial model** of a Horn clause  $C = \bigwedge L_i \rightarrow spt$  if for each valuation  $\beta$ , if all terms in the premises are defined then
  - either all terms in  $L$  are defined and  $L$  true in  $P$
  - if  $s=f(s')$  and  $s', t$  defined then  $f(s')$  defined in  $P$  and  $f(s')\rho t$  true
  - $s'$  is undefined, or  $s$  and  $t$  are undefined.

# Locality and $\mathcal{EL}, \mathcal{EL}^+$

---

**Problem:** Determine if a theory (or a theory extension) is local

**Solution:** Embeddability of partial models into total models

$$\mathcal{T}_1 \text{ } \Psi\text{-local extension of } \mathcal{T}_0 \iff \text{Emb}(\mathcal{T}_0, \mathcal{T}_1)$$

extends [Ganzinger'01, GSW'04, VS'05, VS,Ihlemann'07, Ihlemann,Jacobs,VS'08]

## Examples of local theory extensions related to $\mathcal{EL}$ :

**Fact:** Every semilattice with partial functions, monotone on their (finite) domain of definition (weakly) embeds into a semilattice with total monotone functions.

**Consequence:**  $\text{SLat} \subseteq \text{SLat} \cup \text{Mon}(\Sigma)$  is a local theory extension

# Locality and $\mathcal{EL}, \mathcal{EL}^+$

**Problem:** Determine if a theory (or a theory extension) is local

**Solution:** Embeddability of partial models into total models

$$\mathcal{T}_1 \text{ } \Psi\text{-local extension of } \mathcal{T}_0 \iff \text{Emb}(\mathcal{T}_0, \mathcal{T}_1)$$

extends [Ganzinger'01, GSW'04, VS'05, VS,Ihlemann'07, Ihlemann,Jacobs,VS'08]

## Examples of local theories related to $\mathcal{EL}, \mathcal{EL}^+$ :

There exists a presentation  $SL$  of the theory of semilattices such that:

**Fact:** Every weak partial model of  $SL \cup \text{Mon}(\Sigma)$  (weakly) embeds into a total one

**Consequence:**  $SL \cup \text{Mon}(\Sigma)$  is a local theory

**Fact:** Every Evans partial model  $A$  of  $SL \cup \text{Mon}(\Sigma) \cup Ax$  such that:

for every  $f(g(x)) \leq h(x) \in Ax$  if  $h_A(a)$  is defined then  $g_A(a)$  is defined.

**Consequence:**  $SL \cup \text{Mon}(\Sigma) \cup Ax(RI)$  is  $\Psi$ -stably local,

( $\Psi$  adds  $f_s(u)$  to the set of terms whenever  $f_t(u)$  is in it and  $r \circ s \subseteq t \in RI$  )

# $\mathcal{EL}$ : Hierarchical reasoning

Primitive concepts:	protein, process, substance
Roles:	catalyzes, produces
Terminology: (TBox)	enzyme = protein $\sqcap$ $\exists$ catalyzes.reaction catalyzer = $\exists$ catalyzes.process reaction = process $\sqcap$ $\exists$ produces.substance
Query:	enzyme $\sqsubseteq$ catalyzer?

$\text{SLat} \cup \text{Mon} \models$  enzyme = protein  $\sqcap$  catalyzes-some(reaction)  $\wedge$   
catalyzer = catalyze-some(process)  $\wedge$   
reaction = process  $\sqcap$  produces-some(substance)  
 $\Rightarrow$  enzyme  $\sqsubseteq$  catalyzer

Mon :  $\forall C, D (C \sqsubseteq D \rightarrow \text{catalyze-some}(C) \sqsubseteq \text{catalyze-some}(D))$   
 $\forall C, D (C \sqsubseteq D \rightarrow \text{produces-some}(C) \sqsubseteq \text{produces-some}(D))$

# $\mathcal{EL}$ : Hierarchical reasoning

SLat  $\cup$  Mon  $\wedge$

enzyme = protein  $\sqcap$  catalyzes-some(reaction)  $\wedge$

catalyzer = catalyze-some(process)  $\wedge$

reaction = process  $\sqcap$  produces-some(substance)  $\wedge$

enzyme  $\not\sqsubseteq$  catalyzer

$\models \perp$

$\underbrace{\hspace{15em}}_G$

$G \wedge \text{Mon}$

enzyme = protein  $\sqcap$  catalyzes-some(reaction)  $\wedge$

catalyzer = catalyze-some(process)  $\wedge$

reaction = process  $\sqcap$  produces-some(substance)  $\wedge$

enzyme  $\not\sqsubseteq$  catalyzer

$\forall C, D (C \sqsubseteq D \rightarrow \text{catalyze-some}(C) \sqsubseteq \text{catalyze-some}(D))$

$\forall C, D (C \sqsubseteq D \rightarrow \text{produces-some}(C) \sqsubseteq \text{produces-some}(D))$

# $\mathcal{EL}$ : Hierarchical reasoning

$$\begin{array}{l}
 \text{SLat} \cup \text{Mon} \wedge \\
 \text{enzyme} = \text{protein} \sqcap \text{catalyzes-some}(\text{reaction}) \wedge \\
 \text{catalyzer} = \text{catalyzes-some}(\text{process}) \wedge \\
 \text{reaction} = \text{process} \sqcap \text{produces-some}(\text{substance}) \wedge \\
 \text{enzyme} \not\sqsubseteq \text{catalyzer}
 \end{array} \quad \models \quad \perp$$

$\underbrace{\hspace{15em}}_G$

**Solution 1:** Use  $DPLL(\text{SLat} + \text{UIF})$

$G \wedge \text{Mon}[G]$

$\text{enzyme} = \text{protein} \sqcap \text{catalyzes-some}(\text{reaction})$   
 $\text{catalyzer} = \text{catalyzes-some}(\text{process})$   
 $\text{reaction} = \text{process} \sqcap \text{produces-some}(\text{substance})$   
 $\text{enzyme} \not\sqsubseteq \text{catalyzer}$

$\text{reaction} \triangleright \text{process} \rightarrow \text{catalyzes-some}(\text{reaction}) \triangleright \text{catalyzes-some}(\text{process}), \triangleright \in \{\leq, \geq, =\}$

# $\mathcal{EL}$ : Hierarchical reasoning

$$\begin{array}{l}
 \text{SLat} \cup \text{Mon} \wedge \\
 \text{enzyme} = \text{protein} \sqcap \text{catalyzes-some}(\text{reaction}) \wedge \\
 \text{catalyzer} = \text{catalyze-some}(\text{process}) \wedge \\
 \text{reaction} = \text{process} \sqcap \text{produces-some}(\text{substance}) \wedge \\
 \text{enzyme} \not\sqsubseteq \text{catalyzer}
 \end{array} \quad \models \quad \perp$$

$G$

## Solution 2: Hierarchical reasoning

Base theory (SLat)	Extension
$\text{enzyme} = \text{protein} \sqcap c_1$ $\text{catalyzer} = c_2$ $\text{reaction} = \text{process} \sqcap c_3$ $\text{enzyme} \not\sqsubseteq \text{catalyzer}$ $\text{reaction} \triangleright \text{process} \rightarrow c_1 \triangleright c_2 \quad \triangleright \in \{\leq, \geq, =\}$	$c_1 = \text{catalyzes-some}(\text{reaction})$ $c_2 = \text{catalyzes-some}(\text{process})$ $c_3 = \text{produces-some}(\text{substance})$

Test satisfiability using any prover for SLat (e.g. reduction to SAT)

# $\mathcal{EL}^+$ : Hierarchical reasoning

---

- $\Psi$ -stable locality result

$$\Psi(G) = \text{st}(G) \cup \{f_2(d) \mid f(d) \in \text{st}(G), f_1(f_2(x)) \leq f(x) \in \text{Ax}(RI)\} \cup \dots$$

$\mapsto$  need to take into account more ground terms

**Example:**

$$\text{st}(G) = \{\text{enzyme, protein, reaction, catalyzer, substance} \\ \text{catalyzes-some}(\text{reaction}), \text{helps-producing}(\text{substance}) \}$$

$$\text{Ax}(RI): \text{catalyzes-some}(\text{produces-some}(x)) \leq \text{helps-producing}(x)$$

# $\mathcal{EL}^+$ : Hierarchical reasoning

- $\Psi$ -stable locality result

$$\Psi(G) = \text{st}(G) \cup \{f_2(d) \mid f(d) \in \text{st}(G), f_1(f_2(x)) \leq f(x) \in A_x(RI)\} \cup \dots$$

$\mapsto$  need to take into account more ground terms

**Example:**

$$\text{st}(G) = \{\text{enzyme, protein, reaction, catalyzer, substance} \\ \text{catalyzes-some}(\text{reaction}), \text{helps-producing}(\text{substance})\}$$

$$A_x(RI): \text{catalyzes-some}(\text{produces-some}(x)) \leq \text{helps-producing}(x)$$

$$\text{helps-producing}(\text{substance}) \in \text{st}(G) \Rightarrow \text{produces-some}(\text{substance}) \in \Psi(G)$$

... and (by stable locality) more instances

$\mapsto$  all instances of  $SLUMon \cup A_x(RI)$  with subst. with codomain  $\Psi(G)$

- still in PTIME

# Other examples of local theories

---

Similar locality results also for  $n$ -ary monotone functions:

- $\text{SLat} \subseteq \text{SLat} \cup \text{Mon}(\Sigma)$  is a local theory extension
- There exists a presentation  $SL$  of the theory of semilattices such that  $SL \cup \text{Mon}(\Sigma)$  is a local theory

# Other examples of local theories

---

Similar locality results also for  $n$ -ary monotone functions:

- $\text{SLat} \subseteq \text{SLat} \cup \text{Mon}(\Sigma)$  is a local theory extension
- There exists a presentation  $SL$  of the theory of semilattices such that  $SL \cup \text{Mon}(\Sigma)$  is a local theory

... and for many-sorted domains [Sofronie, Ihlemann 2007]

# Extensions of $\mathcal{EL}$

---

## Extensions of $\mathcal{EL}$ with $n$ -ary roles and numeric domains

**Example 1:** Concepts + maps associating numbers to concepts

### Models:

Set-theoretic:  $(\mathcal{P}(D), \mathbb{R}, \{\cap, \text{MaxCost}\} \cup \{f_r\}_{r \in N_R})$

$\text{MaxCost} : \mathcal{P}(D) \rightarrow \mathbb{R}$ , monotone.

Algebraic:  $(S, \mathbb{R}, \{\wedge, \text{MaxCost}\} \cup \{f_r\}_{r \in N_R})$  where  $(S, \wedge) \in \text{SLat}$ ,  $f_r$  mon.

$\text{MaxCost} : S \rightarrow \mathbb{R}$  monotone

# Extensions of $\mathcal{EL}$

---

## Extensions of $\mathcal{EL}$ with $n$ -ary roles and numeric domains

**Example 2:** Usual concepts + numerical concepts

**Models:**  $(S, \mathcal{P}(\mathbb{R}), \wedge, \{f_r\}_{f \in N_R})$  where  $(S, \wedge) \in \text{SLat}$ ,  $f_r$  mon.

**In general:** Locality  $\mapsto$  parameterized complexity.

PTIME results e.g. when numerical concepts are interpreted over a convex fragment of interval arithmetic ( & TBox in similar fragment)

**Example:**  $(S, \text{Int}(\mathbb{R}), \wedge, \{f_r\}_{f \in N_R})$  where  $(S, \wedge) \in \text{SLat}$ ,  $f_r$  mon.

$\text{Int}(\mathbb{R})$  in the Ord-Horn fragment of Allen's interval arithmetic (translate to Horn clauses over atoms  $x \leq y$ ,  $x = y$ ).

TBoxes: Ord-Horn clauses

# Extensions of $\mathcal{EL}$

**Example 2:** Sorts: **con**, **num**; roles **price**, **has-weight-price**

Interpretation  $\mathcal{I} = (D, \mathbb{R}, \cdot^{\mathcal{I}})$

---

$C$	$\mapsto$	$C^{\mathcal{I}} \subseteq D$
$N$	$\mapsto$	$N^{\mathcal{I}} \subseteq \text{Int}(\mathbb{R})$
		$\uparrow n^{\mathcal{I}} = \{x \in \mathbb{R} \mid x \geq n\}$
		$\downarrow n^{\mathcal{I}} = \{x \in \mathbb{R} \mid x \leq n\}$
		$[n, m]^{\mathcal{I}} = \{x \in \mathbb{R} \mid n \leq x \leq m\}$

---

<b>price</b>	$\mapsto$	<b>price</b> $^{\mathcal{I}} \subseteq D \times \mathbb{R}$
<b>has-weight-price</b>	$\mapsto$	<b>has-weight-price</b> $^{\mathcal{I}} \subseteq D \times \mathbb{R} \times \mathbb{R}$

---

## Complex concepts

$$\exists \text{price}.\uparrow n = \{x \mid \exists k \geq n : \text{price}(x, k)\}$$

the class of all individuals with some price greater than  $n$ .

$$\exists \text{has-weight-price}.\downarrow p = \{x \mid \exists y' = y, \exists p' \leq p \text{ and } \text{has-weight-price}(x, y', p')\}$$

all individuals with weight  $y$  for which a price below  $p$  exist.

# Extensions of $\mathcal{EL}$

Concepts	$car, truck, affordable, C, \downarrow n_1, \uparrow m_1, \downarrow n, \uparrow m$
Roles	$price, has-weight-price, weight$
TBox	$\begin{aligned} \exists price(\downarrow n_1) &\sqsubseteq affordable \\ \exists weight(\uparrow m_1) \sqcap car &\sqsubseteq truck \\ \downarrow n_1 &\sqsubseteq \downarrow n \quad \uparrow m_1 \sqsubseteq \uparrow m \\ C &\sqsubseteq car \\ has-weight-price(\uparrow m, \downarrow n) &\sqsubseteq \exists price(\downarrow n) \sqcap \exists weight(\uparrow m) \\ C &\sqsubseteq \exists has-weight-price(\uparrow m, \downarrow n) \end{aligned}$
Query	$C \sqsubseteq affordable \sqcap truck$

# Extensions of $\mathcal{EL}$

TBox	$\exists \text{price}(\downarrow n_1) \sqsubseteq \text{affordable}$ $\exists \text{weight}(\uparrow m_1) \sqcap \text{car} \sqsubseteq \text{truck}$ $\downarrow n_1 \sqsubseteq \downarrow n \quad \uparrow m_1 \sqsubseteq \uparrow m$ $C \sqsubseteq \text{car}$ $\text{has-weight-price}(\uparrow m, \downarrow n) \sqsubseteq \exists \text{price}(\downarrow n) \sqcap \exists \text{weight}(\uparrow m)$ $C \sqsubseteq \exists \text{has-weight-price}(\uparrow m, \downarrow n)$
Query	$C \sqsubseteq \text{affordable} \sqcap \text{truck}$

## Purification; Use locality

Def	$C_{\text{num}}$	$C_{\text{concept}}$	Mon
$f_{\text{price}}(\downarrow n_1) = c_1$	$\downarrow n \leq \downarrow n_1$	$d_1 \leq \text{affordable}$	$\downarrow n_1 \leq \downarrow n \rightarrow c_1 \leq c$
$f_{\text{price}}(\downarrow n) = c$	$\uparrow m \leq \uparrow m_1$	$d_1 \wedge \text{car} \leq \text{truck}$	$\downarrow n_1 \geq \downarrow n \rightarrow c_1 \geq c$
$f_{\text{weight}}(\uparrow m_1) = d_1$		$e \leq c \wedge d$	$\uparrow m_1 \leq \uparrow m \rightarrow d_1 \leq d$
$f_{\text{weight}}(\uparrow m) = d$		$C \leq \text{car}$	$\uparrow m_1 \geq \uparrow m \rightarrow d_1 \geq d$
$f_{\text{h-w-p}}(\uparrow m, \downarrow n) = e$		$C \leq e$	
		$C \not\leq \text{affordable} \wedge \text{truck}$	

# Extensions of $\mathcal{EL}^+$

---

The results established for  $\mathcal{EL}^+$  extend to  $n$ -ary roles

Role inclusions can be again modeled by axioms at the algebraic level.

$$r_1 \circ r_2 \sqsubseteq r \quad \mapsto \quad \begin{array}{l} \text{If } (x_1, \dots, x_n) \in r_1^{\mathcal{I}} \quad \text{and} \\ (x_n, \dots, x_{n+k}) \in r_2^{\mathcal{I}} \\ \text{then } (x_1, \dots, x_{n-1}, x_{n+1}, \dots, x_{n+k}) \in r^{\mathcal{I}}. \end{array}$$

$$f_{r_1}(x_2, \dots, x_{n-1}, f_{r_2}(x_{n+1}, \dots, x_{n+k})) \subseteq f_r(x_2, \dots, x_{n-1}, x_{n+1}, \dots, x_{n+k}).$$

Stable locality results can also be proved in this case.

# Conclusions

---

- Alternative proof of tractability of  $\mathcal{EL}$ ,  $\mathcal{EL}^+$  based on a notion of locality
  - ↳ a hierarchical reduction to SAT in the case of  $\mathcal{EL}$
  - ↳ reduction to SAT of ground Horn formulae in the case of  $\mathcal{EL}^+$
- Identify tractable extensions of  $\mathcal{EL}$  and  $\mathcal{EL}^+$ 
  - ↳ with  $n$ -ary roles and/or numerical domains.

# Conclusions

---

- Alternative proof of tractability of  $\mathcal{EL}$ ,  $\mathcal{EL}^+$  based on a notion of locality
  - ↳ a hierarchical reduction to SAT in the case of  $\mathcal{EL}$
  - ↳ reduction to SAT of ground Horn formulae in the case of  $\mathcal{EL}^+$
- Identify tractable extensions of  $\mathcal{EL}$  and  $\mathcal{EL}^+$ 
  - ↳ with  $n$ -ary roles and/or numerical domains.

## Future work:

- Which axioms are preserved when embedding partial into total models  
Links with canonical extensions?
- Test our implementation on large  $\mathcal{EL}$  ontologies for comparisons
- Use results on interpolation/interpolant generation in local extensions  
[VS IJCAR'06] for studying modularity issues in distributed ontologies