# RACCOON: A Connection Reasoner for the Description Logic $\mathcal{ALC}$

INT. CONFERENCE ON LOGIC FOR PROGRAMMING, ARTIFICIAL INTELLIGENCE AND REASONING (LPAR 2017)

**Dimas Melo Filho, Fred Freitas**

**Federal University of Pernambuco (CIn/UFPE), Brazil**

**Jens Otten**

**University of Oslo, Norway**

# Summary

- The FOL Connection Method
- The $\mathcal{ALC}$ $\theta$-CM Calculus
  - Representation
  - $\theta$-substitution
  - $Copy$ rule
- RACCOON
- Results & Discussion
- Conclusions & Future Work

# Summary

- The FOL Connection Method
- The $\mathcal{ALC}$ $\theta$-CM Calculus
    - Representation
    - $\theta$-substitution
    - *Copy* rule
- RACCOON
- Results & Discussion
- Conclusions & Future Work

# The Connection Method

- $F_{DNF}$ : $P \lor (\neg P \land Q) \lor (\neg Q \land R) \lor \neg R$

$$\begin{pmatrix} P & \neg P & \neg Q & \neg R \\ & Q & R & \\ & & & \end{pmatrix}$$

- $F_{CNF}$ :
  - $(P \lor \neg P \lor \neg Q \lor \neg R) \land$
  - $(P \lor \neg P \lor R \lor \neg R) \land$
  - $(P \lor Q \lor \neg Q \lor \neg R) \land$
  - $(P \lor Q \lor R \lor \neg R)$

**How to prove it valid then?**

# Validity in DNF Matrices

- $F_{DNF}$ : $P \vee (\neg P \wedge Q) \vee (\neg Q \wedge R) \vee \neg R$
- $\models \begin{pmatrix} P & \neg P & \neg Q & \neg R \\ & Q & R & \end{pmatrix}$ ?
- A formula in DNF is valid when every path has a complimentary pair of literals ($P \vee \neg P$), a connection.All paths must be checked:
  - $\{P, \neg P, \neg Q, \neg R\}$
  - $\{P, \neg P, R, \neg R\}$
  - $\{P, Q, \neg Q, \neg R\}$
  - $\{P, Q, R, \neg R\}$

# FOL CM Example

**Example 1 (DNF, matrix, graphical matrix).** The knowledge base and query

$$\left.\begin{array}{l}\forall x\left(\left(\exists y\, hasPet(x,y)\ \wedge\ Cat(y)\right) \rightarrow CatOwner(x)\right)\\ \forall z\left(OldLady(z) \rightarrow\ \forall k\left(hasPet(z,k) \rightarrow Cat(k)\right)\right)\\ \forall z\left(OldLady(z) \rightarrow\ \exists v\left(hasPet(z,v) \wedge Animal(v)\right)\right)\end{array}\right] \vDash \forall u(OldLady(u) \rightarrow CatOwner(u))$$

# FOL CM Example

**Example 1 (DNF, matrix, graphical matrix).** The knowledge base and query

$$\left.\begin{matrix} \forall x\left(\left(\exists y\ hasPet(x,y)\ \land\ Cat(y)\right) \rightarrow CatOwner(x)\right) \\ \forall z\left(OldLady(z) \rightarrow\ \forall k\left(hasPet(z,k) \rightarrow Cat(k)\right)\right) \\ \forall z\left(OldLady(z) \rightarrow\ \exists v\left(hasPet(z,v) \land Animal(v)\right)\right) \end{matrix}\right] \vDash \forall u(OldLady(u) \rightarrow CatOwner(u))$$

is represented by the matrix (where $a$ is an individual and $f$, a Skolem function):

$\{\{hasPet(x,y),\ \ Cat(y),\ \ \neg CatOwner(x)\},\ \ \{OldLadyu(u),\ \ hasPet(u,v)\ \ ,\neg Cat(v)\},\ \ \{OldLady(z),$ $\neg hasPet(z,f(z))\}, \{OldLady(z), \neg Animal(f(z))\}, \{\neg OldLady(a)\}, \{CatOwner(a)\}\}$

# Proof

$$\begin{bmatrix} \text{hasPet}(x, y) & \text{OldLady}(z) & \text{OldLady}(z) & \text{OldLady}(u) & \neg\text{OldLady}(a) & \text{CatOw}(a) \\ \text{Cat}(y) & \neg\text{hasPet}(z, f(z)) & \neg\text{Anim}(z, f(z)) & \text{hasPet}(u, v) & & \\ \neg\text{CatOw}(x) & & & \neg\text{Cat}(v) & & \end{bmatrix}$$

# Proof

$$\begin{bmatrix} \mathrm{hasPet}(x,y) & \mathrm{OldLady}(z) & \mathrm{OldLady}(z) & \mathrm{OldLady}(u) & \neg\mathrm{OldLady}(a) & \mathrm{CatOw}(a) \\ \mathrm{Cat}(y) & \neg\mathrm{hasPet}(z,f(z)) & \neg\mathrm{Anim}(z,f(z)) & \mathrm{hasPet}(u,v) & & \\ \neg\mathrm{CatOw}(x) & & & \neg\mathrm{Cat}(v) & & \end{bmatrix}$$

$$\sigma = \{x/a\}$$

# Proof

$$\Rightarrow \begin{bmatrix} \mathrm{hasPet}(x,y) & \mathrm{OldLady}(z) & \mathrm{OldLady}(z) & \mathrm{OldLady}(u) & \neg\mathrm{OldLady}(a) & \mathrm{CatOw}(a) \\ \mathrm{Cat}(y) & \neg\mathrm{hasPet}(z,f(z)) & \neg\mathrm{Anim}(z,f(z)) \Rightarrow & \mathrm{hasPet}(u,v) & & \\ \neg\mathrm{CatOw}(x) & & & \neg\mathrm{Cat}(v) & & \end{bmatrix}$$

$$\sigma = \{x/a, y/v\}$$

# Proof

$$\begin{bmatrix} \text{hasPet}(x,y) & \text{OldLady}(z) & \text{OldLady}(z) & \text{OldLady}(u) & \neg\text{OldLady}(a) & \text{CatOw}(a) \\ \text{Cat}(y) & \neg\text{hasPet}(z,f(z)) & \neg\text{Anim}(z,f(z)) & \text{hasPet}(u,v) & & \\ \neg\text{CatOw}(x) & & & \neg\text{Cat}(v) & & \end{bmatrix}$$

$$\sigma = \{x \backslash a, y, v \backslash f(a), u, z \backslash a\}$$

# Summary

- The FOL Connection Method
- The $\mathcal{ALC}$ $\theta$-CM Calculus
  - Representation
  - $\theta$-substitution
  - $Copy$ rule
- RACCOON
- Results & Discussion
- Conclusions & Future Work

# The Description Logic $\mathcal{ALC}$

- Most DLs are decidable fragments of FOL, L2
- Concepts are formed according to the following syntax:

$C ::= A \mid \top \mid \bot \mid C \sqcap D \mid C \sqcup D \mid \neg C \mid \forall r.C \mid \exists r.C,$

  - *A* ranges over concept names, *r* over role names, and *C, D* over concepts
- An $\mathcal{ALC}$ ontology *O* is an ordered pair (*T,A*)
- A TBox T is a set of basic axioms in one of the two forms
  - $C \sqsubseteq D$ *or*
  - $C \equiv D$
- An ABox A w.r.t. a TBox T is a set of assertions of two types
  - *Concept assertions, like C(a)*
  - *Role assertions , like r(a,b)*

# Representation in $\mathcal{ALC}\,\theta$-CM

- Representation without variables, typical from DL

- To eliminate variables, the variable links inside the constructs should be preserved

- *horizontal* lines connect quantified roles to Skolemized concepts
  - Examples: $\forall r.C \sqsubseteq E$ and $E \sqsubseteq \exists r.C$
  - Designated by *r(x,f(x))* and *C(f(x))* in FOL

- *vertical* lines link quantified roles to concepts
  - Examples: $E \sqsubseteq \forall r.C$ and $\exists r.C \sqsubseteq E$
  - Designated by *r(x,y) and C(y)*) in FOL

# $\mathcal{ALC}$ Example

$$\left\{\begin{array}{c} \exists hasPet.Cat \sqsubseteq CatOwner, \\ OldLady \sqsubseteq \exists hasPet.Animal \sqcap \forall hasPet.Cat \end{array}\right\} \vDash OldLady \sqsubseteq CatOwner$$

# $\mathcal{ALC}$ Example

$$\left\{\begin{matrix} \exists hasPet.Cat \sqsubseteq CatOwner, \\ OldLady \sqsubseteq \exists hasPet.Animal \sqcap \forall hasPet.Cat \end{matrix}\right\} \vDash OldLady \sqsubseteq CatOwner$$

$\{\{\underline{hasPet,\ Cat,}\ \neg CatOwner\},\ \{OldLady,\ \underline{hasPet,\ \neg Cat}\},\ \{OldLady,\ \underline{\neg hasPet^1}\},\ \{OldLady,$
$\underline{\neg Animal^1}\}, \{\neg OldLady(a)\}, \{CatOwner(a)\}\}$

# $\mathcal{ALC}$ Example

$$\left\{ \begin{array}{c} \exists hasPet.Cat \sqsubseteq CatOwner, \\ OldLady \sqsubseteq \exists hasPet.Animal \sqcap \forall hasPet.Cat \end{array} \right\} \vDash OldLady \sqsubseteq CatOwner$$

$\{\{\underline{hasPet,\ Cat,}\ \neg CatOwner\},\ \{OldLady,\ \underline{hasPet,\ \neg Cat}\},\ \{OldLady,\ \underline{\neg hasPet^1}\},\ \{OldLady,\ \underline{\neg Animal^1}\},\ \{\neg OldLady(a)\},\ \{CatOwner(a)\}\}$

$$\begin{bmatrix} hasPet & OldLady & OldLady & OldLady & \neg OldLady(a) & CatOwner(a) \\ Cat & \neg hasPet & \neg Animal & hasPet & & \\ \neg CatOwner & & & \neg Cat & & \end{bmatrix}$$

**Figure 1.** The query from Example 1 represented as an $\mathcal{ALC}$ matrix

# $\mathcal{ALC}$ Example

$$\left\{\begin{matrix} \exists hasPet.Cat \sqsubseteq CatOwner, \\ OldLady \sqsubseteq \exists hasPet.Animal \sqcap \forall hasPet.Cat \end{matrix}\right\} \models OldLady \sqsubseteq CatOwner$$

$\{\{\underline{hasPet,\ Cat,}\ \neg CatOwner\},\ \{OldLady,\ \underline{hasPet,\ \neg Cat}\},\ \{OldLady,\ \underline{\neg hasPet^{1}}\},\ \{OldLady,\ \underline{\neg Animal^{1}}\},\ \{\neg OldLady(a)\},\ \{CatOwner(a)\}\}$

$$\begin{bmatrix} hasPet & OldLady & OldLady & OldLady & \neg OldLady(a) & CatOwner(a) \\ Cat & \neg hasPet & \neg Animal & hasPet & & \\ \neg CatOwner & & & \neg Cat & & \end{bmatrix}$$

**Figure 1.** The query from Example 1 represented as an $\mathcal{ALC}$ matrix

$$\begin{bmatrix} hasPet(x,y) & OldLady(z) & OldLady(z) & OldLady(u) & \neg OldLady(a) & CatOw(a) \\ Cat(y) & \neg hasPet(z,f(z)) & \neg Anim(f(z)) & hasPet(u,v) & & \\ \neg CatOw(x) & & & \neg Cat(v) & & \end{bmatrix}$$

# Summary

- The FOL Connection Method
- The $\mathcal{ALC}$ $\theta$-CM Calculus
  - Representation
  - $\theta$-substitution
  - *Copy* rule
- RACCOON
- Results & Discussion
- Conclusions & Future Work

# The $\mathcal{ALC}\,\theta$-CM Calculus

- $\theta$-substitution instead of unification

- $\theta$-substitution is a variable substitution that maps (sometimes implicit) variables to variables or instances.

# $\theta$-substitution - Definitions

- ***The Set of concepts*** $\left(\tau(x)\right)$ [Schmidt & Tishkovsky 2007] of a variable or individual $x$ contains all concepts that were instantiated by $x$
  - $\tau(x) \overset{\text{def}}{=} \{E \in N_C | E(x) \in Path\}$
- ***Skolem condition:*** at most <u>one</u> concept is underlined in the graphical matrix form in a set of concepts or Path for each variable/individual
- $\forall a \left| \{\underline{E^i(a)} \ \in Path\} \right| \leq 1$, where *i* is a column index
- *An instance can be assigned to at most one horizontally linked variable/predicate.*

# FOL simulation and Base Case for $\mathcal{ALC}\ \boldsymbol{\theta}$-CM

- $E \sqsubseteq \exists r.A, \forall r.A \sqsubseteq E \nvdash_{CM} E(a)$

$$\begin{bmatrix} E(x) & E(x) & \neg r(y, g(y)) & A(g(y)) & E(a) \\ \neg r(x, f(x)) & \neg A(f(x)) & \neg E(y) & \neg E(y) & \end{bmatrix}$$

a

- $E \sqsubseteq \exists r.A, \forall r.A \sqsubseteq E \nvdash_{ALC\ \theta-CM} E(a)$, as well:

$$\begin{bmatrix} E & E & \neg r & A & E(a) \\ \neg r & \neg A & \neg E & \neg E & \end{bmatrix}$$

a

# FOL simulation and Base Case for $\mathcal{ALC}$ $\boldsymbol{\theta}$-CM

- $E \sqsubseteq \exists r.A, \forall r.A \sqsubseteq E \nvdash_{CM} E(a)$

$$\begin{bmatrix} E(x) & E(x) & \neg r(y,g(y)) & A(g(y)) & E(a) \\ \neg r(x,f(x)) & \neg A(f(x)) & \neg E(y) & \neg E(y) & \end{bmatrix}$$

X

a

- $E \sqsubseteq \exists r.A, \forall r.A \sqsubseteq E \nvdash_{ALC\ \theta-CM} E(a)$, as well:

$$\begin{bmatrix} E & E & \neg r & A & E(a) \\ \neg r & \neg A & \neg E & \neg E & \end{bmatrix}$$

x X

a

- $Path = \{E(a), \underline{A(x)}, \neg \underline{A(x)}\}$, violating $\forall a \left| \left\{ \underline{E^i} \in N_C \, \middle| \, \underline{E^i(a)} \in Path \right\} \right| \leq 1$

# FOL Connection Calculus - Sequent style

$$Axiom\ (Ax)\quad \frac{}{\{\quad\}, M, Path}$$

$$Start\ Rule\ (St)\quad \frac{C_1, M, \{\}}{\varepsilon, M, \varepsilon}\quad with\ C_1 \in \alpha$$

$$Reduction\ Rule\ (Red)\quad \frac{C, M, Path \cup \{L_2\}}{C \cup \{L_1\}, M, Path \cup \{L_2\}}\ with\ \sigma(L_1) = \sigma(\overline{L_2})$$

$$Extension\ Rule\ (Ext)\ \frac{C_1 \backslash \{L_2\}, M \backslash C_1, Path \cup \{L_1\} \qquad C, M, Path}{C \cup \{L_1\}, M, Path}$$

$$with\ C_1 \in M,\ L_2 \in C_1, \sigma(L_1) = \sigma(\overline{L_2})$$

# The $\mathcal{ALC}\,\theta$-CM Calculus – Sequent Style

$$Axiom\ (Ax)\quad \overline{\{\quad\}, M, Path}$$

$$Start\ Rule\ (St)\quad \frac{C_1, M, \{\}}{\varepsilon, M, \varepsilon}\quad with\ C_1 \in \alpha$$

$$Reduction\ Rule\ (Red)\quad \frac{C, M, Path \cup \{L_2\}}{C \cup \{L_1\}, M, Path \cup \{L_2\}}$$

$$with\ \theta(L_1) = \theta(\overline{L_2})\ and\ the\ Skolem\ condition\ holds$$

$$Extension\ Rule\ (Ext)\ \frac{C_1 \backslash \{L_2\}, M, Path \cup \{L_1\} \qquad C, M, Path}{C \cup \{L_1\}, M, Path}$$

$$with\ C_1 \in M,\ L_2 \in C_1, \theta(L_1) = \theta(\overline{L_2})\ and\ the\ Skolem\ condition\ holds$$

# Example

$$
\cdot \begin{bmatrix}
\text{hasPet} & \text{OldLady} & \text{OldLady} & \text{OldLady} & \neg\text{OldLady(a)} & \text{CatOwner(a)} \\
\text{Cat} & \neg\text{hasPet} & \neg\text{Animal} & \text{hasPet} & & \\
 & & & & & \\
\neg\text{CatOwner} & & & \neg\text{Cat} & &
\end{bmatrix}
$$

$$M = \{\{\underline{h, C}, \neg CO\}, \{O, \underline{h, \neg C}\}, \{O, \underline{\neg h^1}\}, \{O, \underline{\neg A^1}\}, \{\neg O(a)\}, \{CO(a)\}\}$$

$$\frac{}{\varepsilon, M, \varepsilon}\ St$$

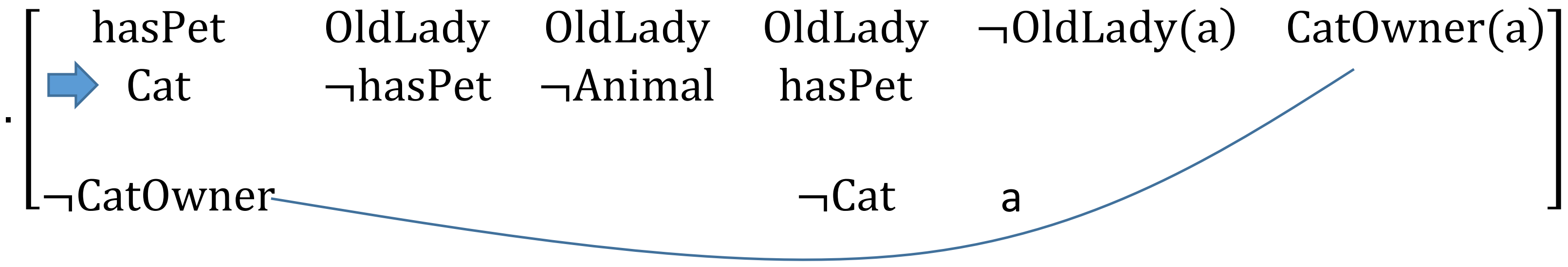
# Example

$$
.\begin{bmatrix}
\text{hasPet} & \text{OldLady} & \text{OldLady} & \text{OldLady} & \neg\text{OldLady(a)} \Rightarrow \text{CatOwner(a)} \\
\text{Cat} & \neg\text{hasPet} & \neg\text{Animal} & \text{hasPet} & \\
 & & & & \\
\neg\text{CatOwner} & & & \neg\text{Cat} &
\end{bmatrix}
$$

$$M = \{\{\underline{h, C}, \neg CO\}, \{O, \underline{h, \neg C}\}, \{O, \underline{\neg h^1}\}, \{O, \underline{\neg A^1}\}, \{\neg O(a)\}, \{CO(a)\}\}$$

$$\frac{\{CO(a)\}, M, \{\}}{\varepsilon, M, \varepsilon}\ St$$

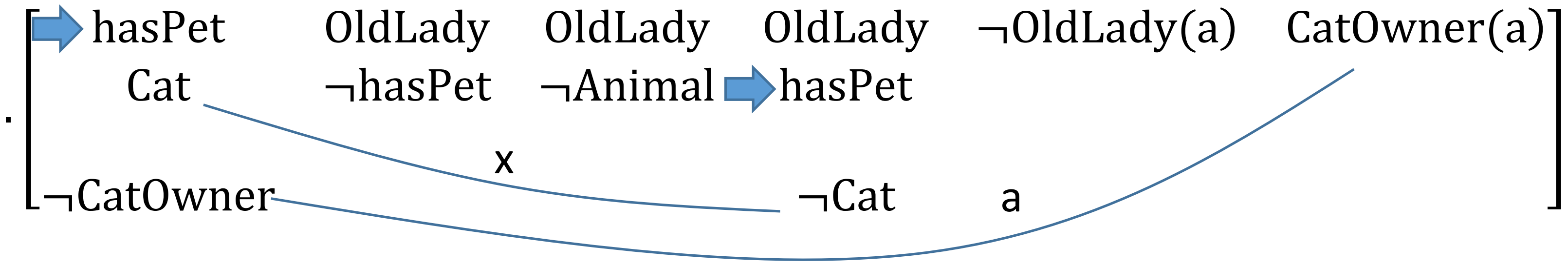
# Example




$$\frac{\dfrac{\{\underline{h(a,x)},\underline{C(x)}\},M,\{CO(a)\} \qquad \{\},M,\{\}}{\{CO(a)\},M,\{\}}\,Ext}{\varepsilon,M,\varepsilon}\,St$$

$$M = \{\{\underline{h, C}, \neg CO\}, \{O, \underline{h, \neg C}\}, \{O, \underline{\neg h^1}\}, \{O, \underline{\neg A^1}\}, \{\neg O(a)\}, \{CO(a)\}\}$$

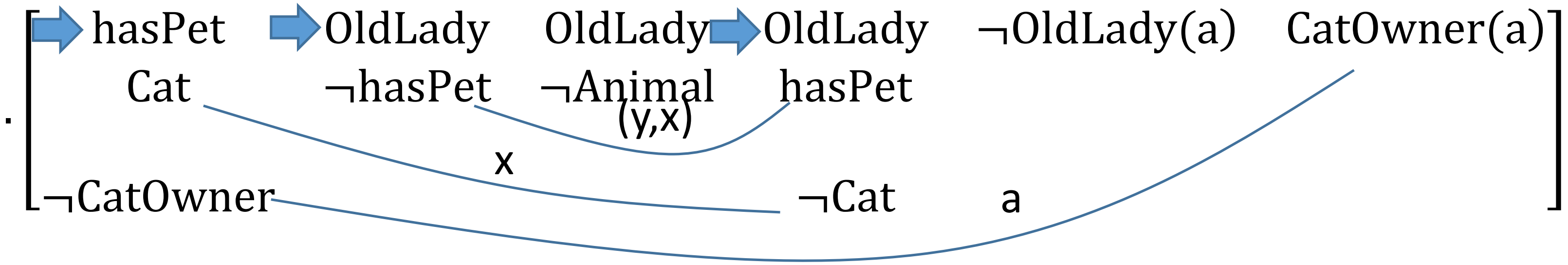
# Example

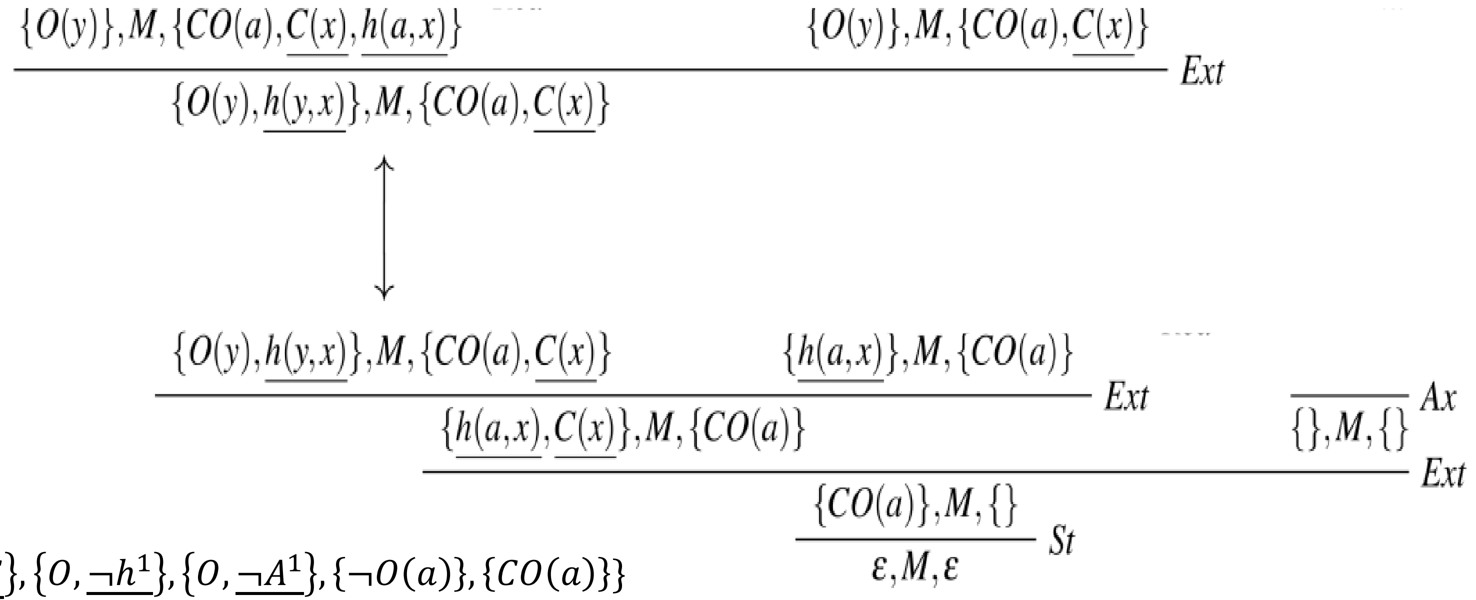



$$\dfrac{\dfrac{\{O(y),\underline{h(y,x)}\},M,\{CO(a),\underline{C(x)}\} \qquad \{\underline{h(a,x)}\},M,\{CO(a)\}}{\{\underline{h(a,x)},\underline{C(x)}\},M,\{CO(a)\}}\,Ext \qquad \dfrac{}{\{\},M,\{\}}\,Ax}{\dfrac{\{CO(a)\},M,\{\}}{\varepsilon,M,\varepsilon}\,St}\,Ext$$

$$M = \{\{\underline{h,C},\neg CO\},\{O,\underline{h,\neg C}\},\{O,\underline{\neg h^1}\},\{O,\underline{\neg A^1}\},\{\neg O(a)\},\{CO(a)\}\}$$

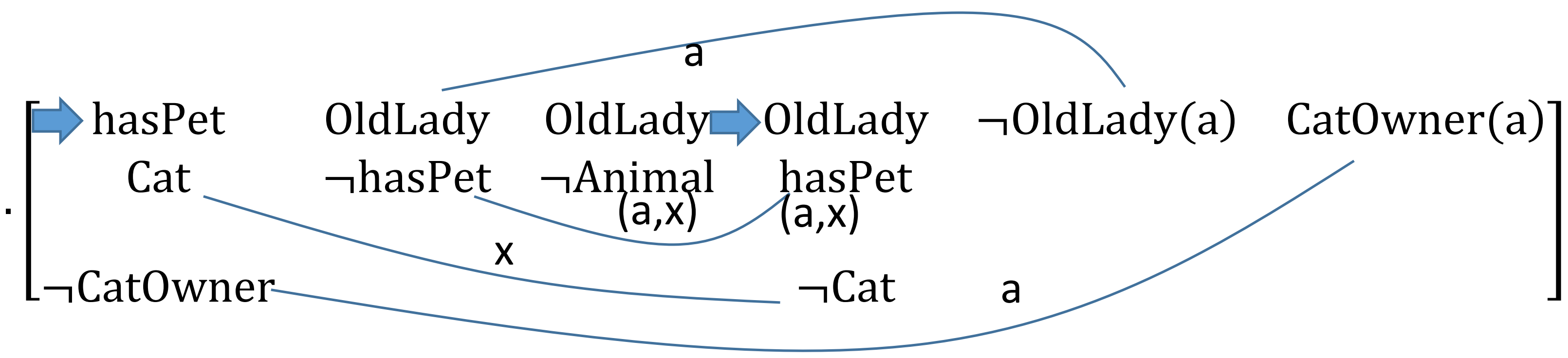
# Example




$M = \{\{\underline{h, C}, \neg CO\}, \{O, \underline{h, \neg C}\}, \{O, \underline{\neg h^1}\}, \{O, \underline{\neg A^1}\}, \{\neg O(a)\}, \{CO(a)\}\}$

# Example



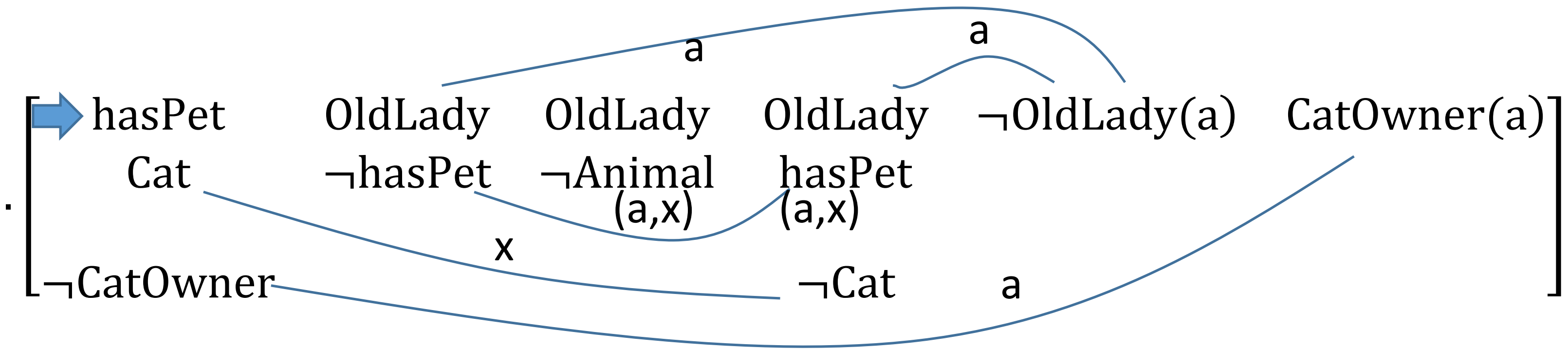

$$\dfrac{\{O(y)\},M,\{CO(a),\underline{C(x)},\underline{h(a,x)}\} \qquad \dfrac{\dfrac{}{\{\},M,\{CO(a),\underline{C(x)},O(a)\}}Ax \qquad \dfrac{}{\{\},M,\{CO(a),\underline{C(x)}\}}Ax}{\{O(y)\},M,\{CO(a),\underline{C(x)}\}}Ext}{\{O(y),\underline{h(y,x)}\},M,\{CO(a),\underline{C(x)}\}}Ext$$

$$\updownarrow$$

$$\dfrac{\dfrac{\dfrac{\{O(y),\underline{h(y,x)}\},M,\{CO(a),\underline{C(x)}\} \qquad \{\underline{h(a,x)}\},M,\{CO(a)\}}{\{\underline{h(a,x)},\underline{C(x)}\},M,\{CO(a)\}}Ext \qquad \dfrac{}{\{\},M,\{\}}Ax}{\{CO(a)\},M,\{\}}Ext}{\varepsilon,M,\varepsilon}St$$

$$M = \{\{\underline{h,C},\neg CO\},\{O,\underline{h,\neg C}\},\{O,\underline{\neg h^1}\},\{O,\underline{\neg A^1}\},\{\neg O(a)\},\{CO(a)\}\}$$

# Example



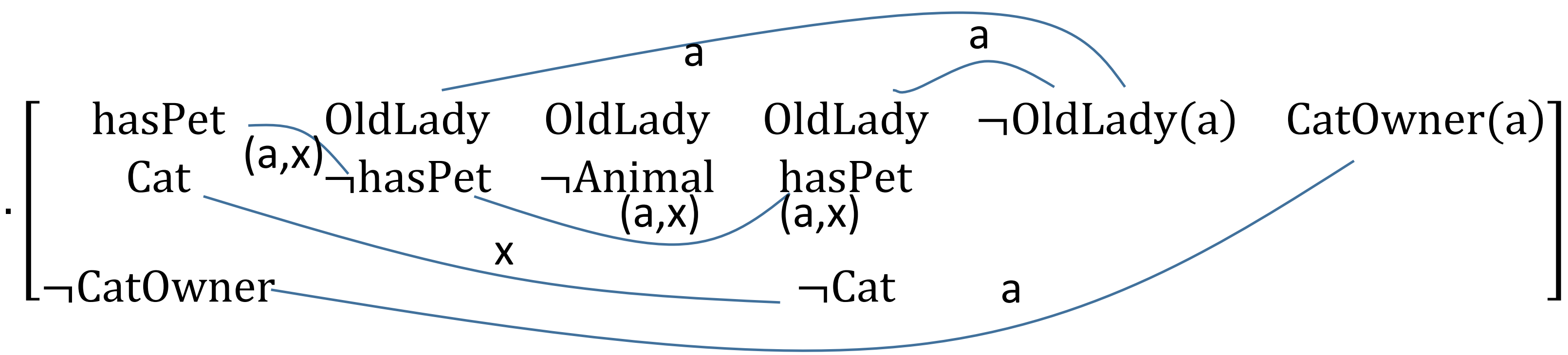

$$\dfrac{\dfrac{\dfrac{}{\{\},M,\{CO(a),\underline{C(x)},\underline{h(a,x)},\neg O(a)\}}\,Ax}{\{O(y)\},M,\{CO(a),\underline{C(x)},\underline{h(a,x)}\}}\,Red \qquad \dfrac{\dfrac{}{\{\},M,\{CO(a),\underline{C(x)},O(a)\}}\,Ax \qquad \dfrac{}{\{\},M,\{CO(a),\underline{C(x)}\}}\,Ax}{\{O(y)\},M,\{CO(a),\underline{C(x)}\}}\,Ext}{\{O(y),\underline{h(y,x)}\},M,\{CO(a),\underline{C(x)}\}}\,Ext$$

$$\updownarrow$$

$$\dfrac{\dfrac{\{O(y),\underline{h(y,x)}\},M,\{CO(a),\underline{C(x)}\} \qquad \{\underline{h(a,x)}\},M,\{CO(a)\}}{\{\underline{h(a,x)},\underline{C(x)}\},M,\{CO(a)\}}\,Ext \qquad \dfrac{}{\{\},M,\{\}}\,Ax}{\dfrac{\{CO(a)\},M,\{\}}{\varepsilon,M,\varepsilon}\,St}\,Ext$$

$$M = \{\{\underline{h, C}, \neg CO\}, \{O, \underline{h, \neg C}\}, \{O, \underline{\neg h^1}\}, \{O, \underline{\neg A^1}\}, \{\neg O(a)\}, \{CO(a)\}\}$$

# Example




$$\frac{\overline{\{\},M,\{CO(a),\underline{C(x)},\underline{h(a,x)},\neg O(a)\}}\;Ax}{\{O(y)\},M,\{CO(a),\underline{C(x)},\underline{h(a,x)}\}}\;Red$$

$$\frac{\overline{\{\},M,\{CO(a),\underline{C(x)},O(a)\}}\;Ax \qquad \overline{\{\},M,\{CO(a),\underline{C(x)}\}}\;Ax}{\{O(y)\},M,\{CO(a),\underline{C(x)}\}}\;Ext$$

$$\frac{\{O(y)\},M,\{CO(a),\underline{C(x)},\underline{h(a,x)}\} \qquad \{O(y)\},M,\{CO(a),\underline{C(x)}\}}{\{O(y),\underline{h(y,x)}\},M,\{CO(a),\underline{C(x)}\}}\;Ext$$

$$\updownarrow$$

$$\frac{\overline{\{\},M,\{CO(a),\underline{\neg h(a,x)^1},\neg O(a)\}}\;Ax}{\{\underline{h(a,x)}\},M,\{CO(a)\}}\;Red$$

$$\frac{\{O(y),\underline{h(y,x)}\},M,\{CO(a),\underline{C(x)}\} \qquad \{\underline{h(a,x)}\},M,\{CO(a)\}}{\{\underline{h(a,x)},\underline{C(x)}\},M,\{CO(a)\}}\;Ext$$

$$\frac{\frac{\{\underline{h(a,x)},\underline{C(x)}\},M,\{CO(a)\} \qquad \overline{\{\},M,\{\}}\;Ax}{\{CO(a)\},M,\{\}}\;Ext}{\varepsilon,M,\varepsilon}\;St$$

$$M=\{\{\underline{h,C},\neg CO\},\{O,\underline{h,\neg C}\},\{O,\underline{\neg h^1}\},\{O,\underline{\neg A^1}\},\{\neg O(a)\},\{CO(a)\}\}$$

# Summary

- The FOL Connection Method
- The $\mathcal{ALC}$ $\theta$-CM Calculus
  - Representation
  - $\theta$-substitution
  - *Copy* rule
- RACCOON
- Results & Discussion
- Conclusions & Future Work

# Blocking

- Usual solution for DL calculi to ensure termination

- For the CM, a new *Copy rule* implements blocking over new instances, to make for the case of cyclic ontologies
- It regulates the generation of new, arbitrary individuals, thus avoiding non-termination

# Blocking rule

$$Copy\ Rule\ (Cop)\ \frac{C \cup \{L_1\}, M \cup \{C_2^{\mu}\}, Path \cup \{L_2\}}{C \cup \{L_1\}, M, Path \cup \{L_2\}}$$

$$with\ L_2 \in C_2^{\mu}, \theta(L_1) = \theta(\overline{L_2}), and\ the\ blocking\ condition\ holds$$

Blocking condition:

- If the set of concepts of new individuals is included in the last generated individual´s set of concepts, then block:
  - $\tau(x_{\mu}^{\theta}) \nsubseteq \tau(x_{\mu-1}^{\theta})$ for not blocking

# Termination Example in $\mathcal{ALC}\boldsymbol{\theta}$-CM

$$Person \sqsubseteq \exists hasFather.Person \nvdash_{\mathcal{ALC}\,\theta-CM} Person(John)$$

$$\begin{bmatrix} Person & Person & & Person(John) \\ \neg hasFather & \neg Person & & \end{bmatrix}$$

# Termination Example in $\mathcal{ALC}\theta$-CM

$$Person \sqsubseteq \exists hasFather.Person \nvdash_{\mathcal{ALC}\theta-CM} \neg Person(John)$$

$$\begin{bmatrix} Person & Person & & Person(John) \\ \neg hasFather & \neg Person & & \end{bmatrix}$$

John

# Termination Example in $\mathcal{ALC}\theta$-CM

$$Person \sqsubseteq \exists hasFather.Person \nvdash_{\mathcal{ALC}\theta-CM} Person(John)$$

$$\begin{bmatrix} Person & Person & {\color{red}Person} & Person(John) \\ \neg hasFather & \neg Person & {\color{red}\neg Person} & \end{bmatrix}$$

John

$$\mu = 1$$

# Termination Example in $\mathcal{ALC}\theta$-CM

$$Person \sqsubseteq \exists hasFather.Person \nvdash_{\mathcal{ALC}\theta-CM} Person(John)$$

$$\begin{bmatrix} Person & Person & Person & Person(John) \\ \neg hasFather & \neg Person & \neg Person & \end{bmatrix}$$

x1

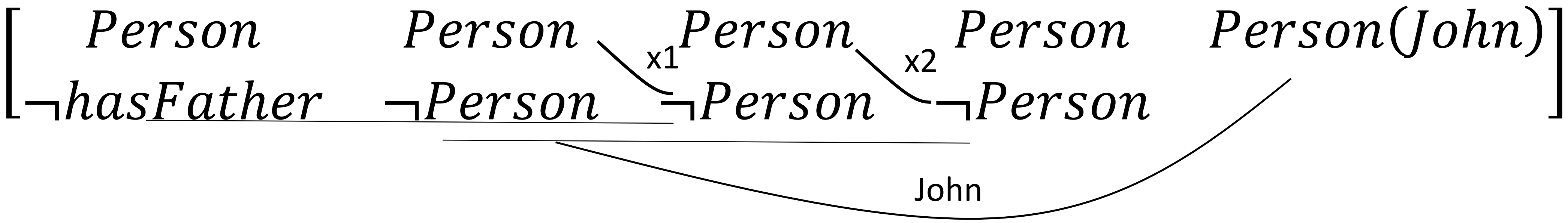
John

$$\mu = 1$$

$$\tau\left(x1_1^\theta\right) = \{\neg Person, Person\}$$

# Termination Example in $\mathcal{ALC}\,\theta$-CM

$$Person \sqsubseteq \exists hasFather.Person \nvdash_{\mathcal{ALC}\,\theta-CM} Person(John)$$




$$\mu = 2$$

$$\tau\left(x1_1^{\theta}\right) = \{\neg Person, Person\}$$

$$\tau\left(x2_2^{\theta}\right) = \{\neg Person, Person\}$$

# Termination Example in $\mathcal{ALC}\boldsymbol{\theta}$-CM

$$Person \sqsubseteq \exists hasFather.Person \nvdash_{\mathcal{ALC}\theta-CM} Person(John)$$

$$\begin{bmatrix} Person & Person & Person & Person & Person(John) \\ \neg hasFather & \neg Person & \neg Person & \neg Person & \end{bmatrix}$$

x1 x2

John

$\mu = 2$

$\tau(x1_1^\theta) = \{\neg Person, Person\}$

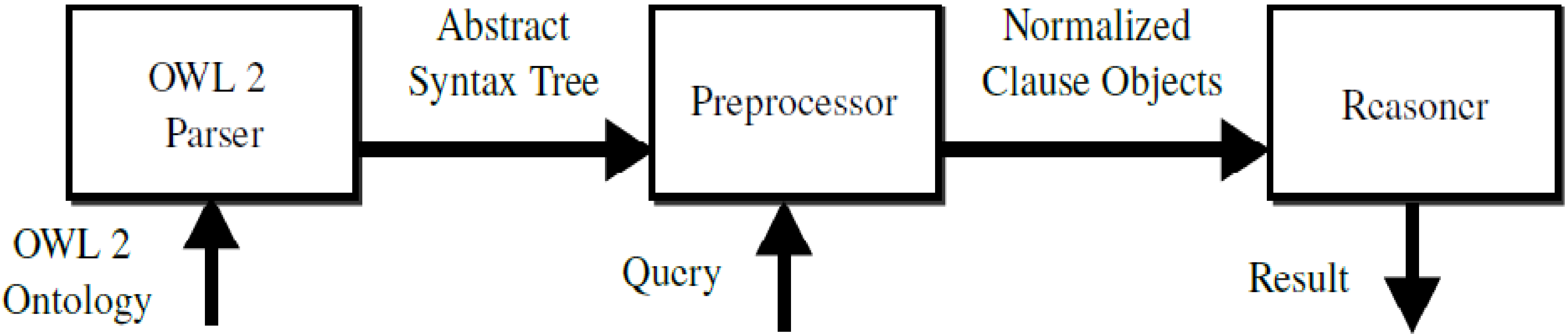
$\tau(x2_2^\theta) = \{\neg Person, Person\}$

Blocked before generating x3, since $\tau(x2_2^\theta) \subseteq \tau(x1_1^\theta)$

# Summary

- The FOL Connection Method
- The $\mathcal{ALC}\,\theta$-CM Calculus
  - Representation
  - $\theta$-substitution
  - $Copy$ rule
- RACCOON
- Results & Discussion
- Conclusions & Future Work

# RACCOON Architecture




- Written in C++ - Only optimization: The regularity condition requires that the active *Path* contains no literals from the extension clause $C_{2,}$ , under unification (Letz et al. 94, Otten 2010)
- $\nexists L_c \in C_{2,}\ L_p \in Path \mid \theta(L_c) = \theta\left(L_p\right).$

# Experiments (based in the ORE)

- ORE competitors:
  - Konclude (Steigmiller et al, 2014)
  - Hermit (Glimm et al, 2014)
  - FACT++ (Tsarkov and Horrocks, 2006)
- Only the consistency task was considered
- Ontologies which had $\mathcal{AL}$, $\mathcal{ALE}$ and $\mathcal{ALC}$ expressivity from ORE's datasets
  - ORE 2014 : 1,621 ontologies
  - ORE 2015 : 401 ontologies
- Timeout : 250s in a 8 GB i5 @ 2.3 GHz, with two cores

# Ontologies x Time: ORE 2014

| #Solved Ontologies by Time Interval | $\mathcal{AL}$ – 825 ontologies | | | | $\mathcal{ALE}$ – 740 ontologies | | | | $\mathcal{ALC}$ – 56 ontologies | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **Raccoon** | **Hermit** | **Fact++** | **Konclude** | **Raccoon** | **Hermit** | **Fact++** | **Konclude** | **Raccoon** | **Konclude** | **Fact++** | **Hermit** |
| **0 - < 0.1s** | 592 | 0 | 0 | 577 | 272 | 0 | 0 | 310 | 46 | 0 | 0 | 47 |
| **0.1 - < 1s** | 201 | 143 | 659 | 195 | 194 | 125 | 393 | 285 | 8 | 26 | 51 | 9 |
| **1 - < 10s** | 13 | 666 | 157 | 44 | 87 | 574 | 323 | 127 | 1 | 30 | 5 | 0 |
| **10 - < 100s** | 7 | 11 | 7 | 9 | 30 | 26 | 4 | 18 | 1 | 0 | 0 | 0 |
| **100 - < 250s** | 0 | 1 | 0 | 0 | 3 | 8 | 3 | 0 | 0 | 0 | 0 | 0 |
| **Timeouts** | **12** | 4 | 2 | 0 | **154** | 7 | 17 | 0 | 0 | 0 | 0 | 0 |
| **Fastest** | **683** | 0 | 0 | 142 | 363 | 0 | 6 | **371** | **45** | 0 | 0 | 11 |

# Ontologies x Time: ORE 2015

| #Solved Ontologies by Time Interval | $\mathcal{AL}$ – 167 ontologies | | | | $\mathcal{ALE}$ – 198 ontologies | | | | $\mathcal{ALC}$ – 36 ontologies | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **Raccoon** | **Hermit** | **Fact++** | **Konclude** | **Raccoon** | **Hermit** | **Fact++** | **Konclude** | **Raccoon** | **Hermit** | **Fact++** | **Konclude** |
| **0 - < 0.1s** | 77 | 0 | 0 | 73 | 55 | 0 | 0 | 60 | 12 | 0 | 0 | 13 |
| **0.1 - < 1s** | 65 | 75 | 91 | 60 | 77 | 52 | 97 | 87 | 12 | 13 | 18 | 17 |
| **1 - < 10s** | 13 | 83 | 63 | 23 | 40 | 123 | 75 | 32 | 7 | 19 | 14 | 3 |
| **10 - < 100s** | 7 | 4 | 8 | 11 | 17 | 6 | 4 | 19 | 4 | 4 | 1 | 3 |
| **100 - < 250s** | 0 | 0 | 0 | 0 | 1 | 10 | 1 | 0 | 0 | 0 | 0 | 0 |
| **Timeouts** | 5 | 5 | 5 | 0 | 8 | 7 | 21 | 0 | 1 | 0 | 3 | 0 |
| **Fastest** | **141** | 0 | 0 | 26 | **143** | 0 | 1 | 54 | **23** | 1 | 0 | 12 |

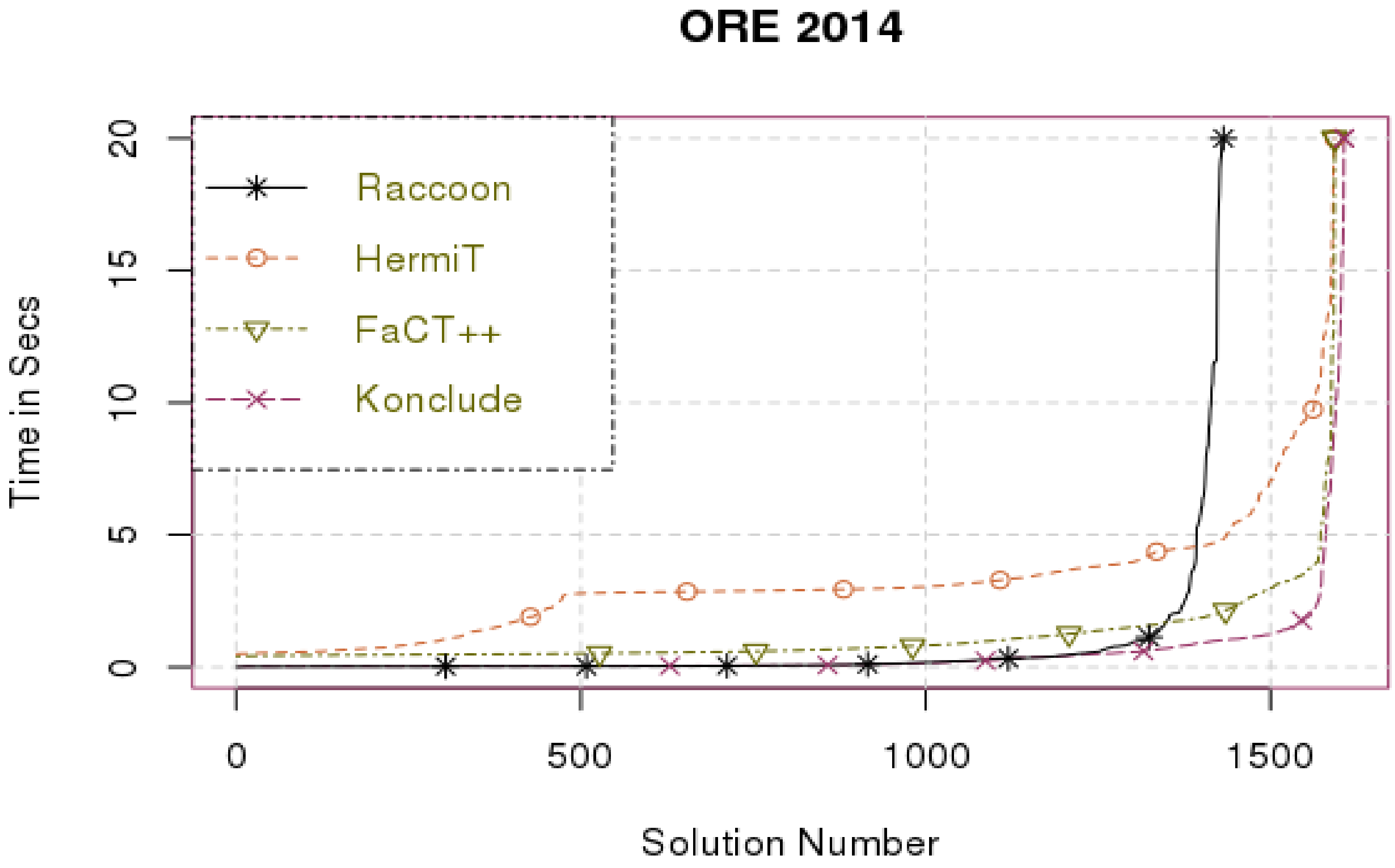
# Average Time x #Axioms : ORE 2014

| Average Time by #axioms | #ontologies | $\mathcal{AL}$ – 825 ontologies | | | | $\mathcal{ALE}$ – 740 ontologies | | | | $\mathcal{ALC}$ – 56 ontologies | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | **Raccoon** | **Hermit** | **Fact++** | **Konclude** | **Raccoon** | **Hermit** | **Fact++** | **Konclude** | **Raccoon** | **Hermit** | **Fact++** | **Konclude** |
| **0-$10^3$ axs** | 837 | 0.03 | 2.75 | 0.49 | 0.04 | 1.90 | 2.36 | 0.57 | 0.05 | 0.03 | 1.70 | 0.49 | 0.04 |
| **$10^3$-$10^4$** | 315 | 0.14 | 4.22 | 0.82 | 0.17 | 2.87 | 2.60 | 0.84 | 0.17 | 8.22 | 2.10 | 0.75 | 0.16 |
| **$10^4$-$10^5$** | 427 | 0.58 | 3.53 | 1.64 | 0.86 | 1.55 | 4.99 | 3.97 | 0.99 | 1.66 | 3.49 | 1.19 | 0.38 |
| **$10^5$-$10^6$** | 39 | 14.80 | 25.58 | 20.91 | 27.99 | 15.65 | 66.63 | 16.65 | 15.32 | - | - | - | - |
| **$\geq 10^6$axs** | 3 | N/A | 51.94 | N/A | 40.66 | 41.44 | 53.67 | 53.73 | 58.73 | - | - | - | - |
| **Average** | - | 0.30 | 3.34 | 0.94 | 0.55 | 2.79 | 5.60 | 2.45 | 1.21 | 0.79 | 1.92 | 0.58 | 0.08 |

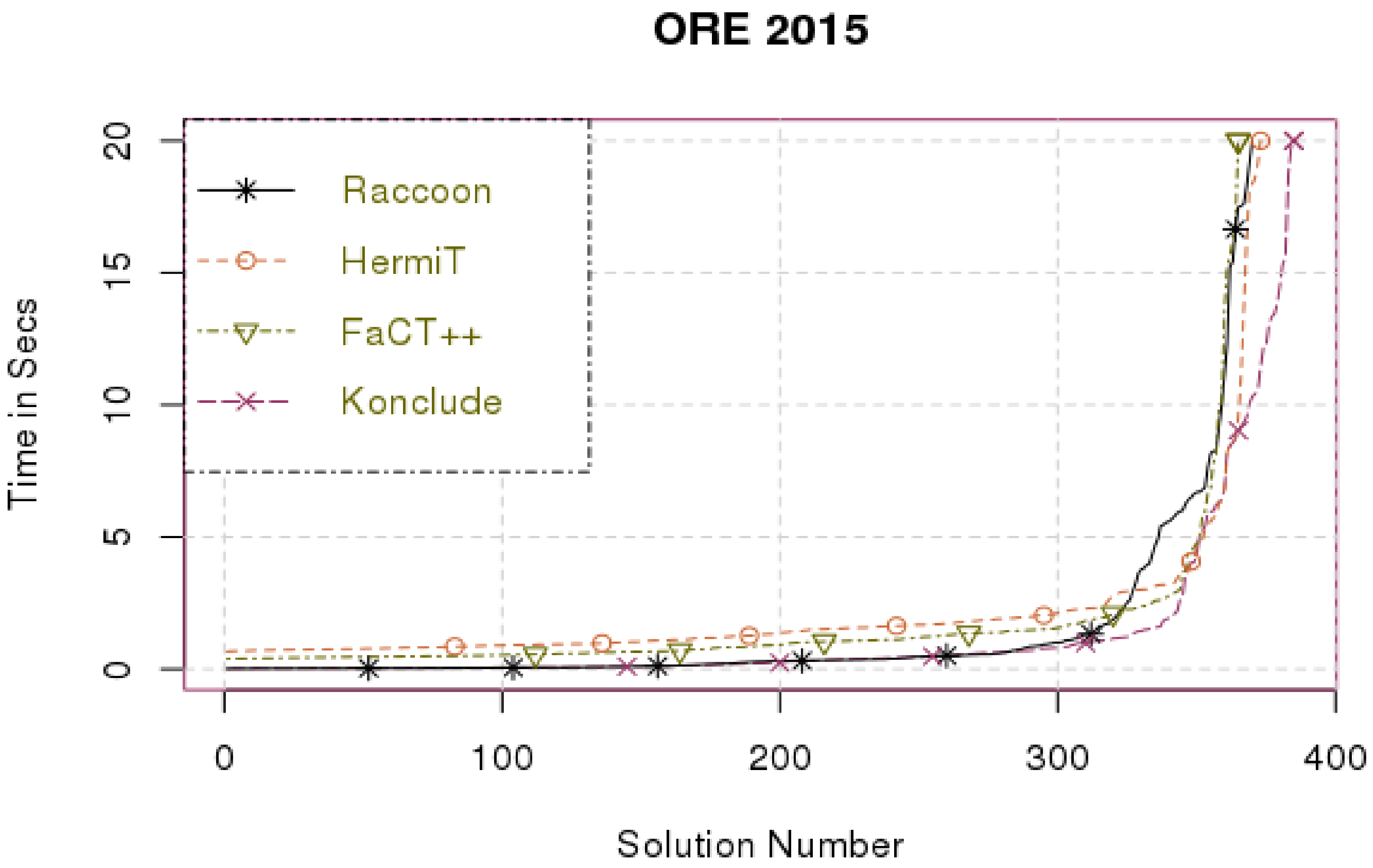
# Average Time x #Axioms : ORE 2015

| Average Time by #axioms | #ontologies | $\mathcal{AL}$ – 167 ontologies | | | | $\mathcal{ALE}$ – 198 ontologies | | | | $\mathcal{ALC}$ – 36 ontologies | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | **Raccoon** | **Hermit** | **Fact++** | **Konclude** | **Raccoon** | **Hermit** | **Fact++** | **Konclude** | **Raccoon** | **Hermit** | **Fact++** | **Konclude** |
| **0-$10^3$ axs** | 119 | 0.03 | 0.80 | 0.47 | 0.04 | 1.53 | 0.91 | 0.57 | 0.06 | 0.50 | 0.82 | 0.49 | 0.04 |
| **$10^3$-$10^4$** | 96 | 0.12 | 1.09 | 0.73 | 0.15 | 7.67 | 1.25 | 0.77 | 0.22 | 10.94 | 1.17 | 0.82 | 0.19 |
| **$10^4$-$10^5$** | 136 | 0.56 | 2.16 | 1.61 | 0.84 | 0.89 | 3.97 | 5.48 | 1.24 | 1.49 | 1.75 | 1.33 | 0.51 |
| **$10^5$-$10^6$** | 48 | 11.43 | 13.86 | 26.30 | 21.45 | 10.07 | 84.00 | 9.26 | 16.81 | 11.42 | 15.74 | 13.89 | 15.19 |
| **$\geq 10^6$axs** | 2 | N/A | 41.80 | N/A | 38.72 | - | - | - | - | 25.71 | 41.67 | N/A | 33.48 |
| **Average** | - | 1.28 | 2.39 | 2.83 | 2.49 | 4.20 | 11.60 | 2.84 | 2.94 | 3.76 | 4.03 | 2.09 | 2.86 |

# Behavior w.r.t. Ontology Size



# Behavior w.r.t. Ontology Size



# Results - Conclusions

- RACCOON had a quite large number of timeouts (166) in the ORE 2014
  - But less than FACT++ in the ORE 2015
- RACCOON and Konclude were almost always the fastest in all segments
  - It performs better for small ontologies
- This performance is likely due to the efficient parsing and regularity
- RACCOON runs over $\mathcal{ALC}$, while the other solvers can deal with larger DLs

# Summary

- The FOL Connection Method
- The $\mathcal{ALC}$ $\theta$-CM Calculus
  - Representation
  - $\theta$-substitution
  - *Copy* rule
- RACCOON
- Results & Discussion
- Conclusions & Future Work

# Conclusions

- $\mathcal{ALC}\,\theta$-CM is a connection method for the DL $\mathcal{ALC}$
    - Replaces Skolem functions and unification by $\theta$-substitutions
    - Blocking scheme, with a new *Copy rule*, to assure termination
- Implemented in RACCOON:https://github.com/dmfilho/raccoon
    - Implementation can be competitive
    - Fast for $\mathcal{ALC}$ and smaller DLs

# Conclusions

- Initial motivation, basic research for DL reasoning
  - Trying out established, efficient FOL methods in the DL setting
- RACCOON is the implementation of a specially-tailored DL connection calculus
  - The first of its kind: DL+ connection calculus
- Promising performance
  - No typical DL optimization
  - Good for small ontologies, common place in the Semantic Web

# Practical Future Work

- DL optimizations
- High number of timeouts will be tackled in a future version
- Implementing the other two ORE tasks, realization and subsumption
- Implementing (in)equality, DL cardinality restrictions
  - $\geq / \leq n\, r$ for $\mathcal{ALCN}$ and $\geq / \leq n\, r.C$ for $\mathcal{SHQ}$ and $\mathcal{EL}$++
  - Enables an ORE participation in the $\mathcal{EL}$ segment
  - Already theoretically solved, but not implemented and tested yet
- More expressive constructs & DLs

# Theoretical Future Work

- Encompassing techniques such as dynamic and double blocking (Horrocks and Sattler, 1999) for inverse roles
- Dealing with nominals
- …

# Thank you for your attention!

- Questions?
- Suggestions?

# Unification and $\theta$-substitution are equivalent

| Unification | | $\theta$-substitution | |
|---|---|---|---|
| **Input** | **Output** | **Input** | **Output** |
| $L_1 = E(x)$<br>$L_2 = \neg E(a)$ | $\sigma(L_1) = E(a)$ | $L_1 = E \; or \; E(x)$<br>$L_2 = \neg E(a)$ | $\theta(L_1) = E(a)$ |
| $L_1 = E(x)$<br>$L_2 = \neg E(y)$ | $\sigma(L_1) = E(y)$ | $L_1 = E \; or \; E(x)$<br>$L_2 = \neg E \; or \neg E(y)$ | $\theta(L_1) = E(y)$ |
| $L_1 = E(b)$<br>$L_2 = \neg E(a)$ | Not unifiable | $L_1 = E(b)$<br>$L_2 = \neg E(a)$ | No $\theta$-substitution:<br>Not unifiable |
| $L_1 = E(x)$<br>$L_2 = \neg E(f(y))$ | $\sigma(L_1) = \{E(f(y))\}$ | $L_1 = E$<br>$L_2 = \underline{\neg E^i}$ | $\theta(L_1) = E(y)$<br>$\tau(y) = \tau(y) \cup \{\underline{\neg E^i}\}$ |
| $L_1 = E(g(x))$<br>$L_2 = \neg E(f(y))$ | Not unifiable | $L_1 = \underline{E}^k$<br>$L_2 = \underline{\neg E^j}$ | No $\theta$-substitution,as Skolem Condition does not hold: $\forall a$<br>$\left\| \left\{ \underline{E^i} \middle\| \underline{E^i(a)} \in Path \right\} \right\| \leq 1$ |

# Inductive Case for $\mathcal{ALC}\,\theta$-CM


Figure 7. Tentative $\mathcal{ALC}\,\theta$-CM connection proof for the inductive case

# Inductive Case for $\mathcal{ALC}\,\theta$-CM


**Figure 7. Tentative $\mathcal{ALC}\,\theta$-CM connection proof for the inductive case**

$Path = \{E(a), \underline{A(a)}, D(a), \ldots, B(a), \neg\underline{C(a)}\}$

# Inductive Case for $\mathcal{ALC}\,\theta$-CM


**Figure 7.** Tentative $\mathcal{ALC}\,\theta$-CM connection proof for the inductive case

$Path = \{E(a)\underline{, A(a)}, D(a), \ldots, B(a), \neg\underline{C(a)}\}$

violating $\forall a \left| \left\{ \underline{E^i} \in N_C \; \middle| \underline{E^i(a)} \in Path \right\} \right| \leq 1$

# Inductive Case for $\mathcal{ALC}\,\theta$-CM

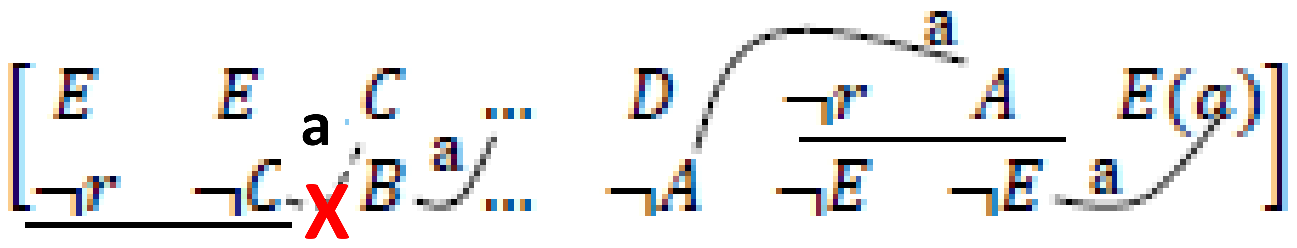



**Figure 7.** Tentative $\mathcal{ALC}\,\theta$-CM connection proof for the inductive case

$$Path = \{E(a), \underline{A(a)}, D(a), \dots, B(a), \neg\underline{C(a)}\}$$

violating $\forall a \left| \left\{ \underline{E^i} \in N_C \;\middle|\; \underline{E^i(a)} \in Path \right\} \right| \leq 1$

# Problematic Ontologies

$$
\begin{bmatrix}
\cdots & Z & Z & Z & Z & \cdots & Z & Z & \neg B_0 & \neg B_1 & \cdots & \neg B_n \\
 & r & r & r & r & & r & r & & & & \\
 & \neg r & \neg Z & \neg r & \neg Z & & \neg r & \neg Z & & & & \\
\hline
 & B_0 & B_0 & B_1 & B_1 & \cdots & B_n & B_n & & & &
\end{bmatrix}
$$

# Algorithm

```
func proveLiteral (clause, literalIndex, path) {
    if literalindex >= clause.size()                    \\ check if clause literals exhausted
        return true;
    if path.contains (literalindex)                     \\ regularity: if literal is already in the path
        return false;
    if path.containsNeg (literalindex)                  \\ Reduction rule
        return proveLiteral (clause, literalIndex+1, path)   \\ try next literal
    path.push (literalIndex);                           \\ put literal on path
    for each connection in literal.connections {        \\ try each literal's possible connection
        if connection.valid (path) {                    \\ if we prove with the current connection
            path.pop ();                                \\ remove literal from path to try next literal
            if proveLiteral (clause, literalIndex+1, path)   \\ and try the clause's next literal
                return true;
            path.push (literalIndex)                    \\ if failed, push the literal back again
        }                                               \\ and try next connection (backtrack)
    }
    path.pop ();
    return false;                                       \\ no connection worked; not proven
}
```