Classifying Chemicals Using Description Graphs and Logic Programming

Despoina Magka, Boris Motik and Ian Horrocks

Department of Computer Science, University of Oxford

May 28, 2012

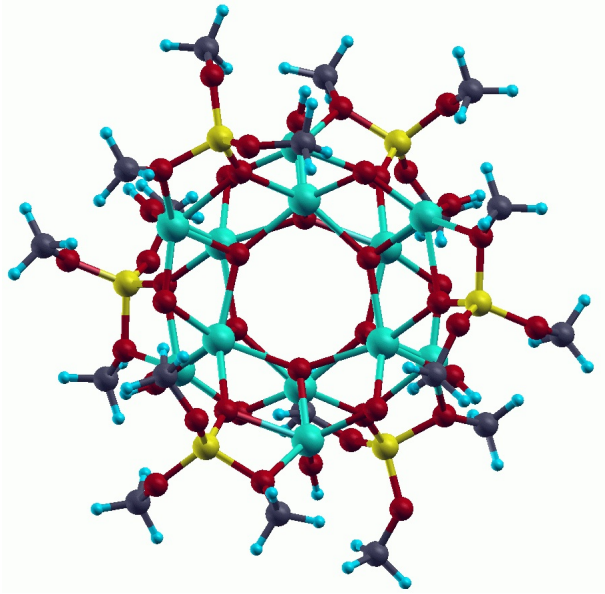
Outline

Modelling Chemicals with OWL

- OWL used for the representation of molecular structures

Modelling Chemicals with OWL

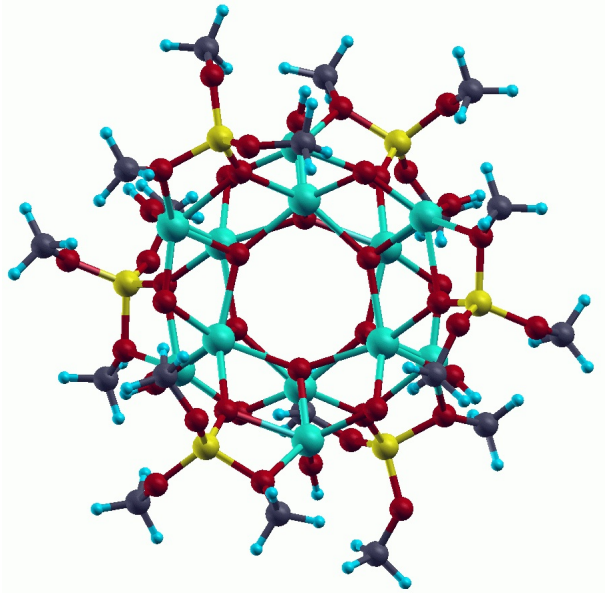
- OWL used for the representation of molecular structures



- Classification of chemical compounds
  [Villanueva-Rosales & Dumontier, OWLED 2007]

Modelling Chemicals with OWL

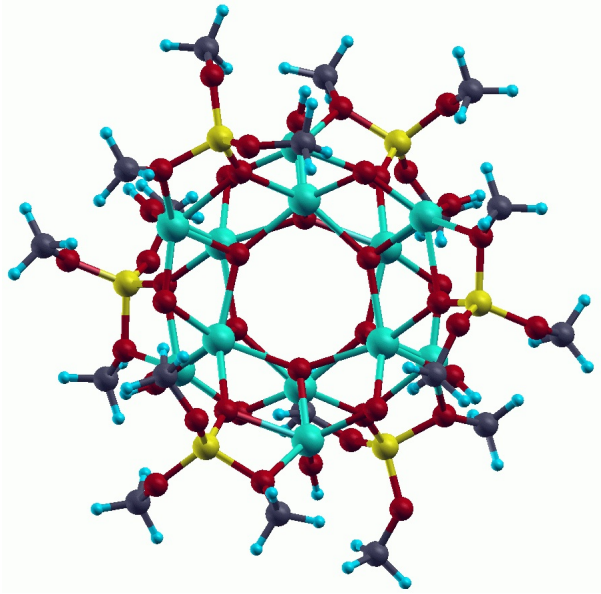
- OWL used for the representation of molecular structures



- Classification of chemical compounds
  [Villanueva-Rosales & Dumontier, OWLED 2007]
- Chemical information integration
  [Konyk et al., DILS, 2008]

Modelling Chemicals with OWL

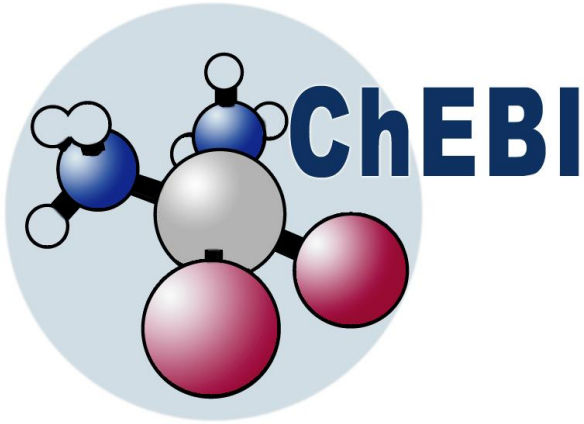
- OWL used for the representation of molecular structures



- Classification of chemical compounds
  [Villanueva-Rosales & Dumontier, OWLED 2007]
- Chemical information integration
  [Konyk et al., DILS, 2008]
- Subsumptions between molecules and chemical classes
  [Hastings et al., OWLED, 2010]

THE CHEBI ONTOLOGY

- OWL ontology Chemical Entities of Biological Interest



EMBL-EBI

The ChEBI Ontology

- OWL ontology Chemical Entities of Biological Interest
  - Freely accessible dictionary of ‘small’ molecular entities

The ChEBI Ontology

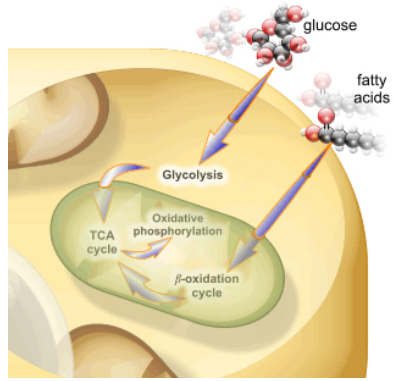
- OWL ontology Chemical Entities of Biological Interest
  - Freely accessible dictionary of ‘small’ molecular entities
  - High quality annotation and taxonomy of chemicals

The ChEBI Ontology

- OWL ontology Chemical Entities of Biological Interest
  - Freely accessible dictionary of ‘small’ molecular entities
  - High quality annotation and taxonomy of chemicals
  - Interoperability between researchers

The ChEBI Ontology

- OWL ontology Chemical Entities of Biological Interest
  - Freely accessible dictionary of ‘small’ molecular entities
  - High quality annotation and taxonomy of chemicals
  - Interoperability between researchers
  - Drug discovery and elucidation of metabolic pathways



AUTOMATE CHEMICAL CLASSIFICATION

- ChEBI is manually incremented

Automate Chemical Classification

- ChEBI is manually incremented
- Currently contains approx. 28,000 fully annotated entities

Automate Chemical Classification

- ChEBI is manually incremented
- Currently contains approx. 28,000 fully annotated entities
- Grows at a rate of ~1,500 entities per curator per year

Automate Chemical Classification

- ChEBI is manually incremented
- Currently contains approx. 28,000 fully annotated entities
- Grows at a rate of ~1,500 entities per curator per year
- Biologically interesting entities possibly > 1,000,000

Automate Chemical Classification

- ChEBI is manually incremented
- Currently contains approx. 28,000 fully annotated entities
- Grows at a rate of ~1,500 entities per curator per year
- Biologically interesting entities possibly > 1,000,000
- Each new molecule is subsumed by several chemical classes

Automate Chemical Classification

- ChEBI is manually incremented
- Currently contains approx. 28,000 fully annotated entities
- Grows at a rate of ~1,500 entities per curator per year
- Biologically interesting entities possibly > 1,000,000
- Each new molecule is subsumed by several chemical classes
  - Is dinitrogen inorganic?

Automate Chemical Classification

- ChEBI is manually incremented
- Currently contains approx. 28,000 fully annotated entities
- Grows at a rate of ~1,500 entities per curator per year
- Biologically interesting entities possibly > 1,000,000
- Each new molecule is subsumed by several chemical classes
  - Is dinitrogen inorganic?
  - Does cyclobutane contain a four-membered ring?

Automate Chemical Classification

- ChEBI is manually incremented
- Currently contains approx. 28,000 fully annotated entities
- Grows at a rate of ~1,500 entities per curator per year
- Biologically interesting entities possibly > 1,000,000
- Each new molecule is subsumed by several chemical classes
  - Is dinitrogen inorganic?
  - Does cyclobutane contain a four-membered ring?
  - Is acetylene a hydrocarbon?

Automate Chemical Classification

- ChEBI is manually incremented
- Currently contains approx. 28,000 fully annotated entities
- Grows at a rate of ~1,500 entities per curator per year
- Biologically interesting entities possibly > 1,000,000
- Each new molecule is subsumed by several chemical classes
  - Is dinitrogen inorganic?
  - Does cyclobutane contain a four-membered ring?
  - Is acetylene a hydrocarbon?
  - Does benzaldehyde contain a benzene ring?

Automate Chemical Classification

- ChEBI is manually incremented
- Currently contains approx. 28,000 fully annotated entities
- Grows at a rate of ~1,500 entities per curator per year
- Biologically interesting entities possibly > 1,000,000
- Each new molecule is subsumed by several chemical classes
  - Is dinitrogen inorganic?
  - Does cyclobutane contain a four-membered ring?
  - Is acetylene a hydrocarbon?
  - Does benzaldehyde contain a benzene ring?
- Speed up curating tasks with automated reasoning tools

Automate Chemical Classification

- ChEBI is manually incremented
- Currently contains approx. 28,000 fully annotated entities
- Grows at a rate of ~1,500 entities per curator per year
- Biologically interesting entities possibly > 1,000,000
- Each new molecule is subsumed by several chemical classes
  - Is dinitrogen inorganic? ⇝ Yes
  - Does cyclobutane contain a four-membered ring? ⇝ Yes
  - Is acetylene a hydrocarbon? ⇝ Yes
  - Does benzaldehyde contain a benzene ring? ⇝ Yes
- Speed up curating tasks with automated reasoning tools

(MIS)REPRESENTING RINGS WITH OWL

- Chemical compounds with rings are highly frequent

(Mis)Representing Rings with OWL

- Chemical compounds with rings are highly frequent
- Fundamental inability of OWL to represent cycles

(Mis)Representing Rings with OWL

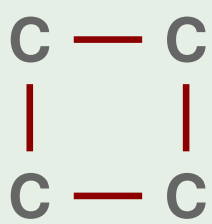
- Chemical compounds with rings are highly frequent
- Fundamental inability of OWL to represent cycles
- At least one tree-shaped model for each consistent OWL knowledge base

(Mis)Representing Rings with OWL

- Chemical compounds with rings are highly frequent
- Fundamental inability of OWL to represent cycles
- At least one tree-shaped model for each consistent OWL knowledge base

Example

$$\mathsf{Cyclobutane} \sqsubseteq \exists(=4)\mathsf{hasAtom}.(\mathsf{Carbon} \sqcap \exists(=2)\mathsf{hasBond}.\mathsf{Carbon})$$

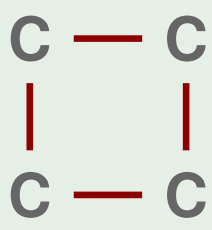


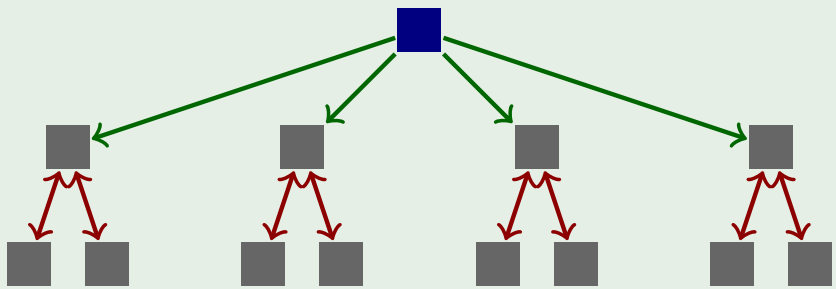
(Mis)Representing Rings with OWL

- Chemical compounds with rings are highly frequent
- Fundamental inability of OWL to represent cycles
- At least one tree-shaped model for each consistent OWL knowledge base

Example

$$\text{Cyclobutane} \sqsubseteq \exists(=4)\text{hasAtom}.(\text{Carbon} \sqcap \exists(=2)\text{hasBond}.\text{Carbon})$$



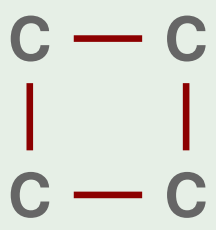


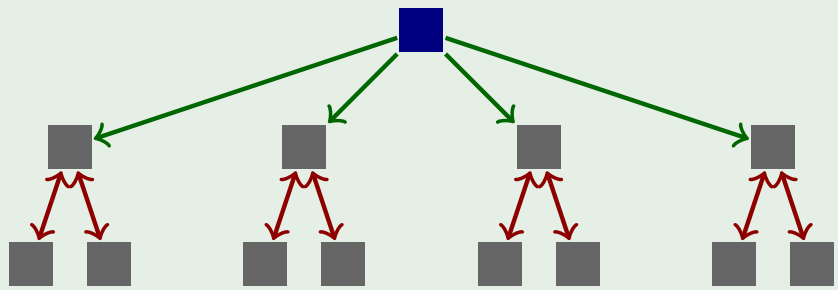
(Mis)Representing Rings with OWL

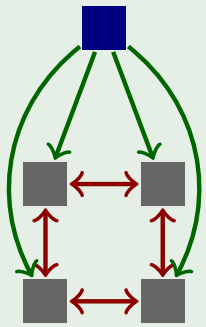
- Chemical compounds with rings are highly frequent
- Fundamental inability of OWL to represent cycles
- At least one tree-shaped model for each consistent OWL knowledge base

Example

$$\text{Cyclobutane} \sqsubseteq \exists(=4)\text{hasAtom}.(\text{Carbon} \sqcap \exists(=2)\text{hasBond}.\text{Carbon})$$





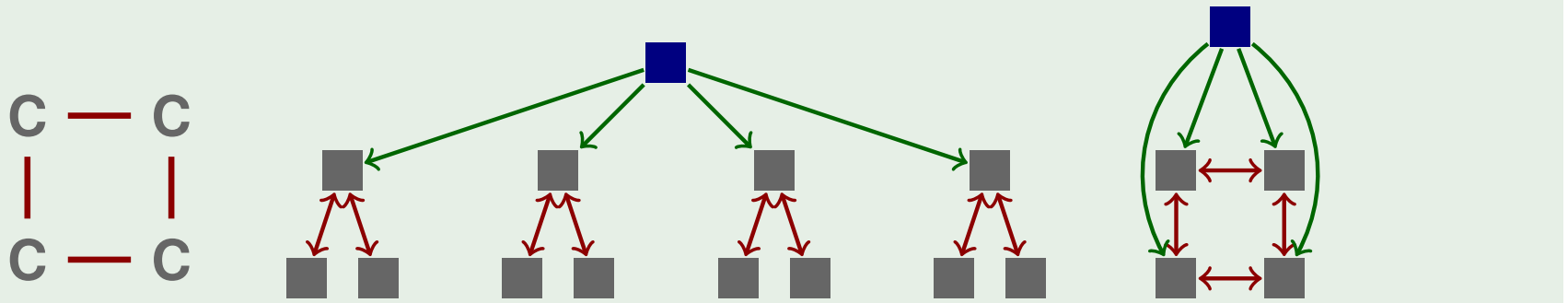


(Mis)Representing Rings with OWL

- Chemical compounds with rings are highly frequent
- Fundamental inability of OWL to represent cycles
- At least one tree-shaped model for each consistent OWL knowledge base

Example

$$\text{Cyclobutane} \sqsubseteq \exists(=4)\text{hasAtom}.(\text{Carbon} \sqcap \exists(=2)\text{hasBond}.\text{Carbon})$$



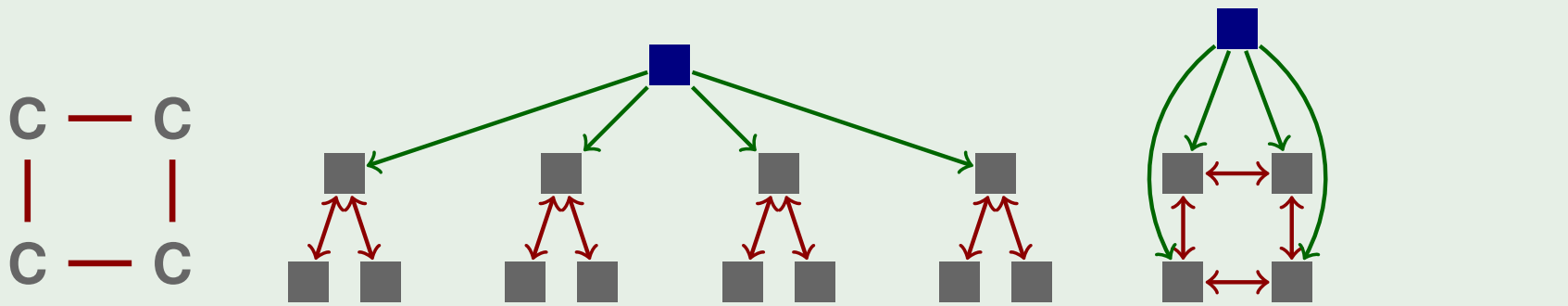
- OWL-based reasoning support

(Mis)Representing Rings with OWL

- Chemical compounds with rings are highly frequent
- Fundamental inability of OWL to represent cycles
- At least one tree-shaped model for each consistent OWL knowledge base

Example

$$\text{Cyclobutane} \sqsubseteq \exists(=4)\text{hasAtom}.(\text{Carbon} \sqcap \exists(=2)\text{hasBond}.\text{Carbon})$$



- OWL-based reasoning support
  - Does cyclobutane contain a four-membered ring? ✘
  - Does benzaldehyde contain a benzene ring? ✘

OWL Extensions

- Limitation of OWL to represent cycles (partially) remedied by extension of OWL with Description Graphs and rules [Motik et al., 2009]

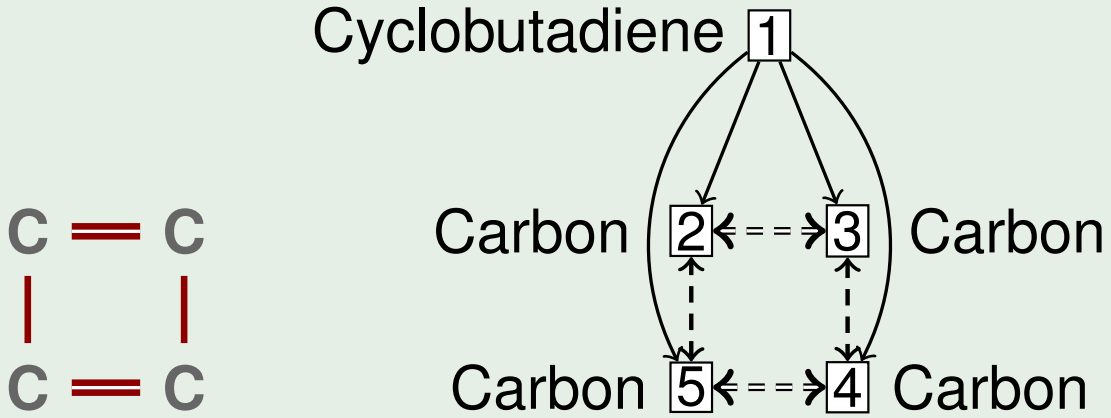
OWL Extensions

- Limitation of OWL to represent cycles (partially) remedied by extension of OWL with Description Graphs and rules [Motik et al., 2009]
- A Description Graph represents structures by means of a directed labeled graph

OWL Extensions

- Limitation of OWL to represent cycles (partially) remedied by extension of OWL with Description Graphs and rules [Motik et al., 2009]
- A Description Graph represents structures by means of a directed labeled graph

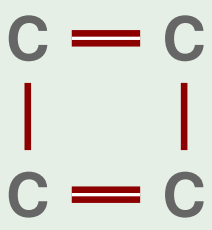
Example

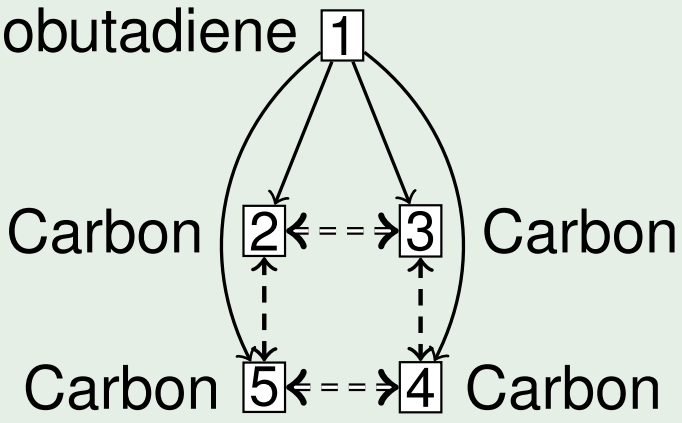


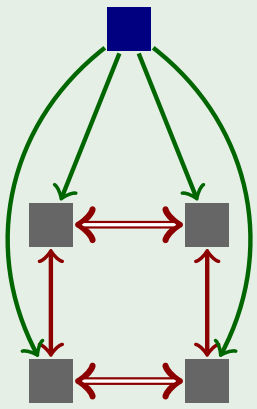
OWL Extensions

- Limitation of OWL to represent cycles (partially) remedied by extension of OWL with Description Graphs and rules [Motik et al., 2009]
- A Description Graph represents structures by means of a directed labeled graph

Example



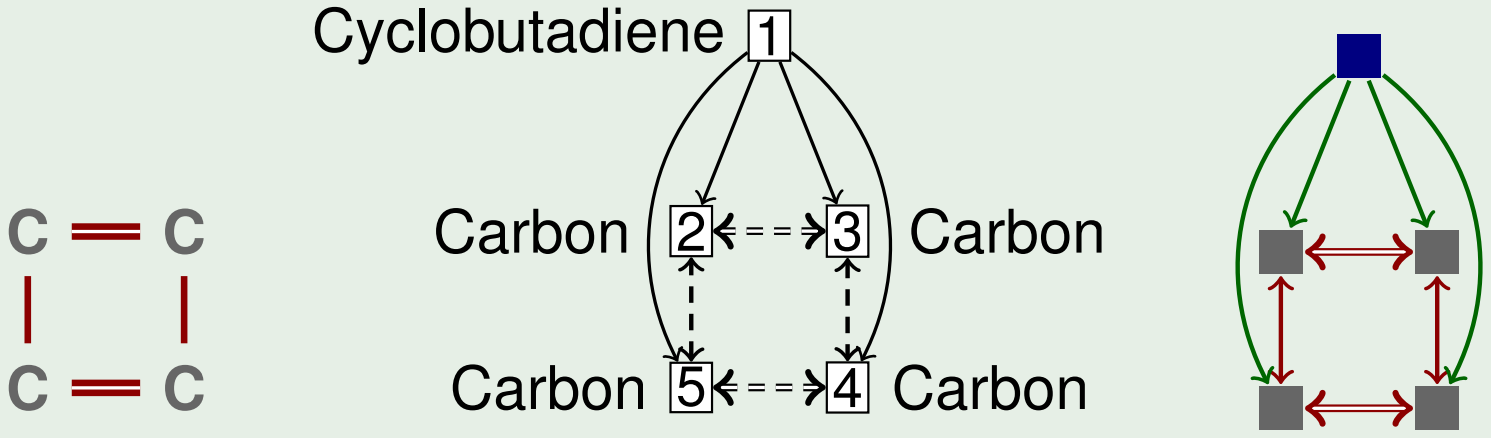




OWL Extensions

- Limitation of OWL to represent cycles (partially) remedied by extension of OWL with Description Graphs and rules [Motik et al., 2009]
- A Description Graph represents structures by means of a directed labeled graph

Example

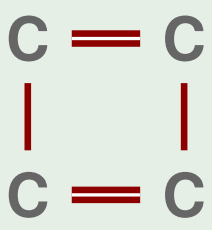


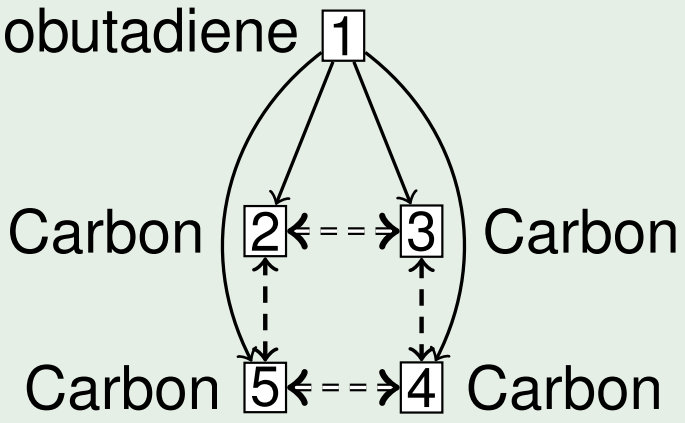
- Does cyclobutadiene have a conjugated four-membered ring?

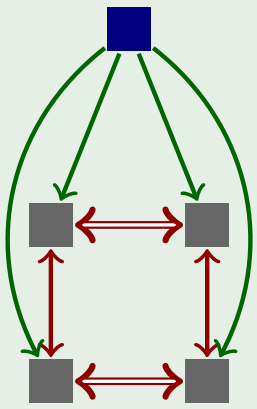
OWL Extensions

- Limitation of OWL to represent cycles (partially) remedied by extension of OWL with Description Graphs and rules [Motik et al., 2009]
- A Description Graph represents structures by means of a directed labeled graph

Example





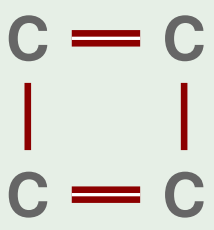


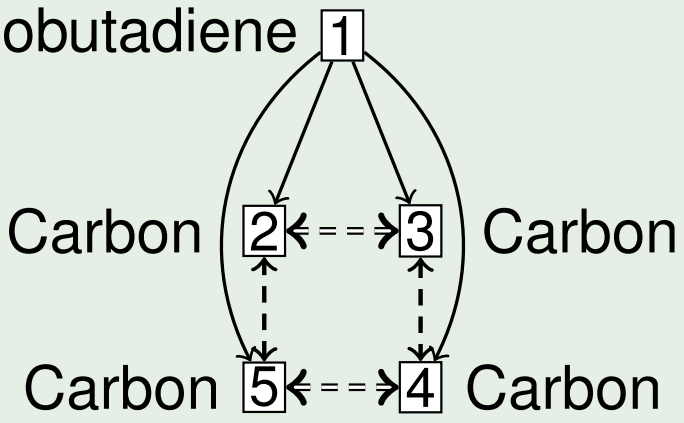
- Does cyclobutadiene have a conjugated four-membered ring? ✓

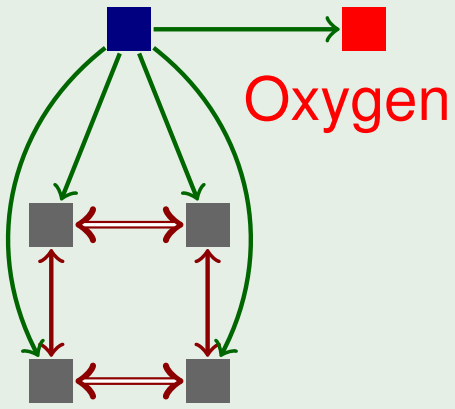
OWL Extensions

- Limitation of OWL to represent cycles (partially) remedied by extension of OWL with Description Graphs and rules [Motik et al., 2009]
- A Description Graph represents structures by means of a directed labeled graph

Example



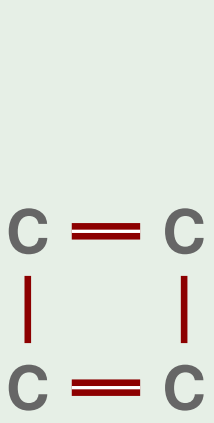


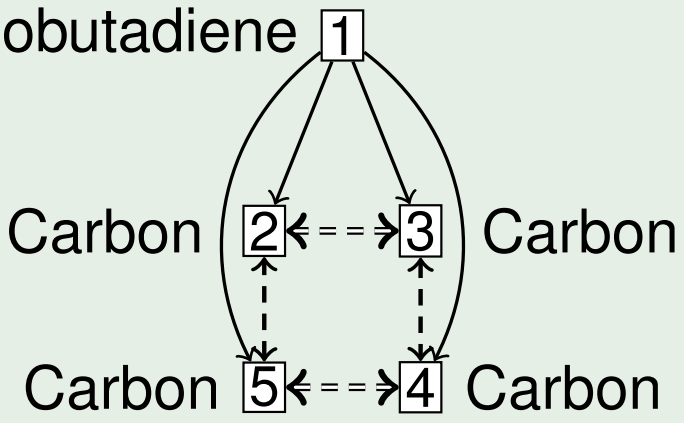


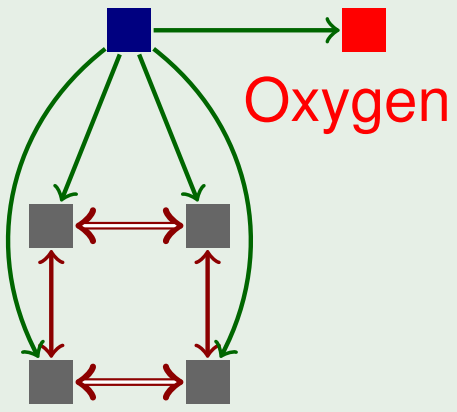
OWL Extensions

- Limitation of OWL to represent cycles (partially) remedied by extension of OWL with Description Graphs and rules [Motik et al., 2009]
- A Description Graph represents structures by means of a directed labeled graph

Example





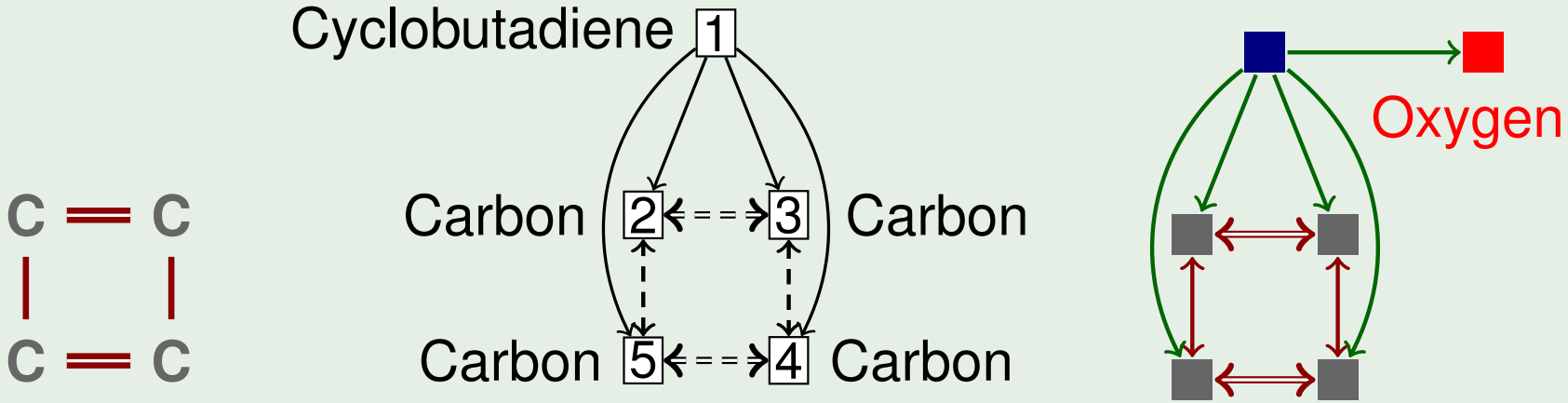


$\forall \text{hasAtom}.(\text{Carbon} \sqcup \text{Hydrogen}) \sqsubseteq \text{Hydrocarbon}$

OWL Extensions

- Limitation of OWL to represent cycles (partially) remedied by extension of OWL with Description Graphs and rules [Motik et al., 2009]
- A Description Graph represents structures by means of a directed labeled graph

Example



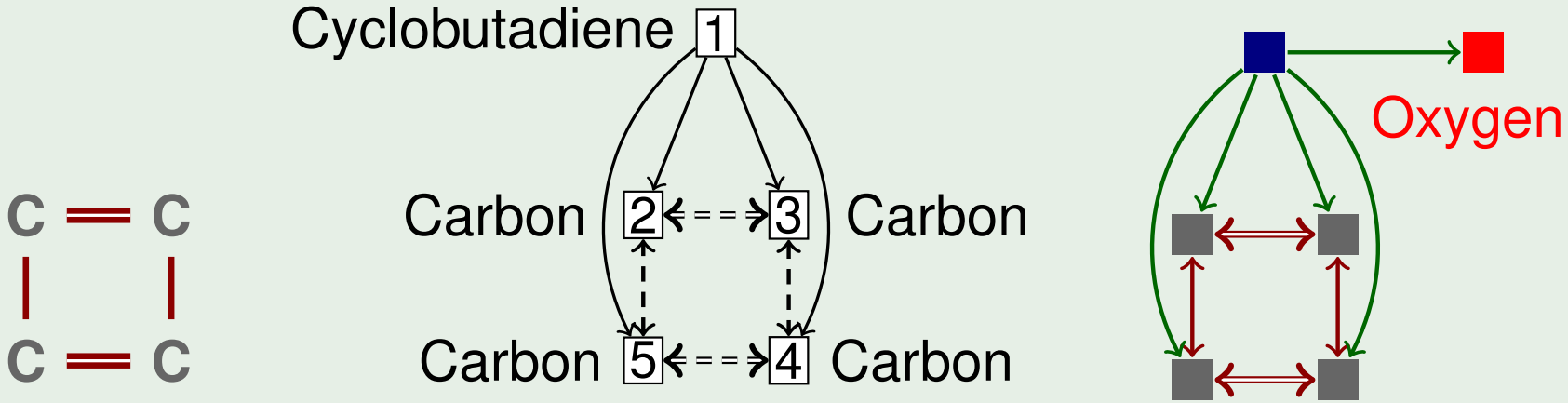
$\forall \text{hasAtom}.(\text{Carbon} \sqcup \text{Hydrogen}) \sqsubseteq \text{Hydrocarbon}$

- Is cyclobutadiene a hydrocarbon?

OWL Extensions

- Limitation of OWL to represent cycles (partially) remedied by extension of OWL with Description Graphs and rules [Motik et al., 2009]
- A Description Graph represents structures by means of a directed labeled graph

Example



$\forall \text{hasAtom}.(\text{Carbon} \sqcup \text{Hydrogen}) \sqsubseteq \text{Hydrocarbon}$

- Is cyclobutadiene a hydrocarbon? ✘

Results Overview

- Key idea:

Results Overview

- Key idea:
  - Switch from first-order logic to logic programming semantics

Results Overview

- Key idea:
  - Switch from first-order logic to logic programming semantics
  - Replace classical negation with negation-as-failure

Results Overview

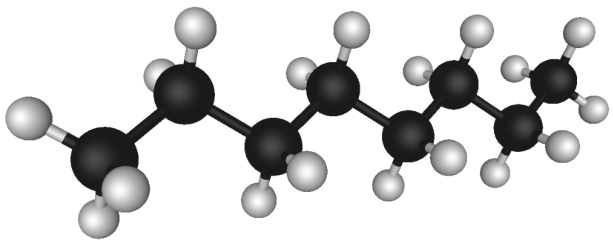
- Key idea:
  - Switch from first-order logic to logic programming semantics
  - Replace classical negation with negation-as-failure
- Double benefit:

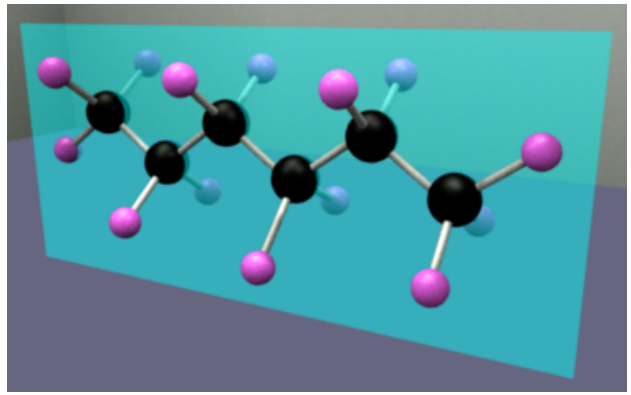
Results Overview

- Key idea:
  - Switch from first-order logic to logic programming semantics
  - Replace classical negation with negation-as-failure
- Double benefit:
  1. Encode chemical classes based on the absence of information

Results Overview

- Key idea:
  - Switch from first-order logic to logic programming semantics
  - Replace classical negation with negation-as-failure
- Double benefit:
  1. Encode chemical classes based on the absence of information (e.g. hydrocarbons, inorganic molecules, saturated compounds,. . .)



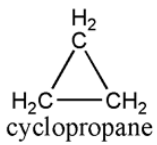


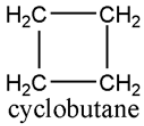
Results Overview

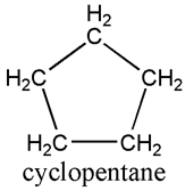
- Key idea:
  - Switch from first-order logic to logic programming semantics
  - Replace classical negation with negation-as-failure
- Double benefit:
  1. Encode chemical classes based on the absence of information (e.g. hydrocarbons, inorganic molecules, saturated compounds,...)
  2. Represent rings with adequate precision

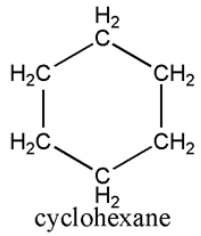
Results Overview

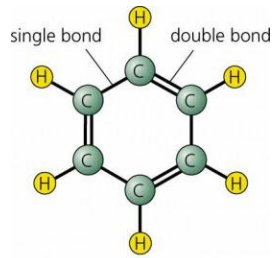
- Key idea:
  - Switch from first-order logic to logic programming semantics
  - Replace classical negation with negation-as-failure
- Double benefit:
  1. Encode chemical classes based on the absence of information (e.g. hydrocarbons, inorganic molecules, saturated compounds,. . .)
  2. Represent rings with adequate precision (e.g. cycloalkanes, benzene rings,. . .)











Results Overview

- Key idea:
  - Switch from first-order logic to logic programming semantics
  - Replace classical negation with negation-as-failure
- Double benefit:
  1. Encode chemical classes based on the absence of information (e.g. hydrocarbons, inorganic molecules, saturated compounds,...)
  2. Represent rings with adequate precision (e.g. cycloalkanes, benzene rings,...)
- Expressive decidable logic-based formalism for modelling structured entities: Description Graph Logic Programs (DGLPs)

Results Overview

- Key idea:
  - Switch from first-order logic to logic programming semantics
  - Replace classical negation with negation-as-failure
- Double benefit:
  1. Encode chemical classes based on the absence of information (e.g. hydrocarbons, inorganic molecules, saturated compounds,...)
  2. Represent rings with adequate precision (e.g. cycloalkanes, benzene rings,...)
- Expressive decidable logic-based formalism for modelling structured entities: Description Graph Logic Programs (DGLPs)
- Prototypical implementation of a logic-based chemical classification software

Outline

What is a DGLP Ontology?

- The syntactic objects of a DGLP ontology:

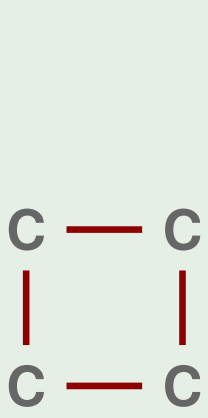
What is a DGLP Ontology?

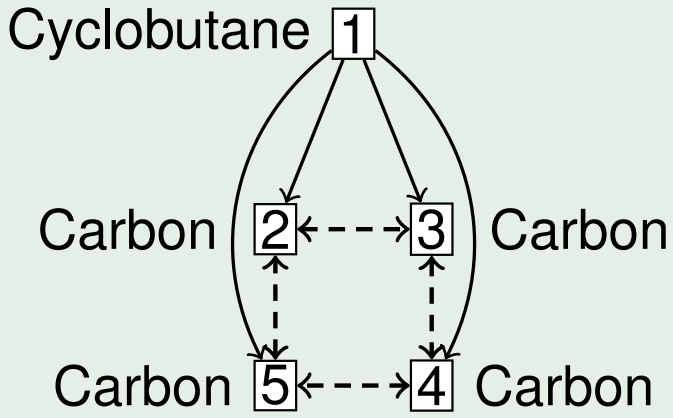
- The syntactic objects of a DGLP ontology:
  - Description graphs

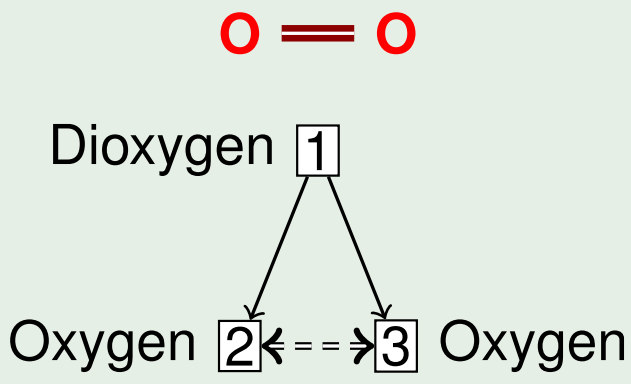
What is a DGLP Ontology?

- The syntactic objects of a DGLP ontology:
  - Description graphs

Example







What is a DGLP Ontology?

- The syntactic objects of a DGLP ontology:
  - Description graphs
  - Function-free FOL Horn rules

What is a DGLP Ontology?

- The syntactic objects of a DGLP ontology:
  - Description graphs
  - Function-free FOL Horn rules

Example

$$\begin{aligned} \text{Bond}(x, y) &\rightarrow \text{Bond}(y, x) \\ \text{SingleBond}(x, y) &\rightarrow \text{Bond}(x, y) \end{aligned}$$

What is a DGLP Ontology?

- The syntactic objects of a DGLP ontology:
  - Description graphs
  - Function-free FOL Horn rules

Example

$$\begin{aligned} \text{Bond}(x, y) &\rightarrow \text{Bond}(y, x) \\ \text{SingleBond}(x, y) &\rightarrow \text{Bond}(x, y) \end{aligned}$$

  - Rules with negation-as-failure

What is a DGLP Ontology?

- The syntactic objects of a DGLP ontology:
  - Description graphs
  - Function-free FOL Horn rules

Example

$$\begin{aligned} \text{Bond}(x, y) &\rightarrow \text{Bond}(y, x) \\ \text{SingleBond}(x, y) &\rightarrow \text{Bond}(x, y) \end{aligned}$$

  - Rules with negation-as-failure

Example

$$\begin{aligned} \text{HasAtom}(x, y) \wedge \text{Carbon}(y) &\rightarrow \text{HasCarbon}(x) \\ \text{Molecule}(x) \wedge \textbf{not}\ \text{HasCarbon}(x) &\rightarrow \text{Inorganic}(x) \end{aligned}$$

What is a DGLP Ontology?

- The syntactic objects of a DGLP ontology:
  - Description graphs
  - Function-free FOL Horn rules

Example

$$\begin{aligned} \text{Bond}(x, y) &\rightarrow \text{Bond}(y, x) \\ \text{SingleBond}(x, y) &\rightarrow \text{Bond}(x, y) \end{aligned}$$

  - Rules with negation-as-failure

Example

$$\begin{aligned} \text{HasAtom}(x, y) \wedge \text{Carbon}(y) &\rightarrow \text{HasCarbon}(x) \\ \text{Molecule}(x) \wedge \textbf{not}\ \text{HasCarbon}(x) &\rightarrow \text{Inorganic}(x) \end{aligned}$$

  - Facts

What is a DGLP Ontology?

- The syntactic objects of a DGLP ontology:
  - Description graphs
  - Function-free FOL Horn rules

Example

$$\begin{aligned} \text{Bond}(x, y) &\rightarrow \text{Bond}(y, x) \\ \text{SingleBond}(x, y) &\rightarrow \text{Bond}(x, y) \end{aligned}$$

  - Rules with negation-as-failure

Example

$$\begin{aligned} \text{HasAtom}(x, y) \wedge \text{Carbon}(y) &\rightarrow \text{HasCarbon}(x) \\ \text{Molecule}(x) \wedge \textbf{not}\ \text{HasCarbon}(x) &\rightarrow \text{Inorganic}(x) \end{aligned}$$

  - Facts

Example

$$\text{Cyclobutane}(c_1), \quad \text{Dinitrogen}(c_2), \ldots$$

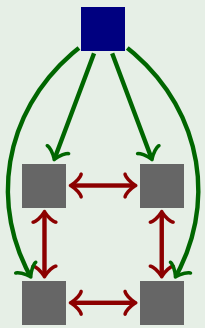
Encoding Description Graphs

- Translate DGs into logic programs with function symbols

Encoding Description Graphs

- Translate DGs into logic programs with function symbols

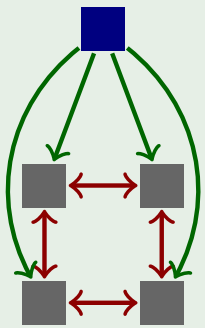
Example



Encoding Description Graphs

- Translate DGs into logic programs with function symbols

Example

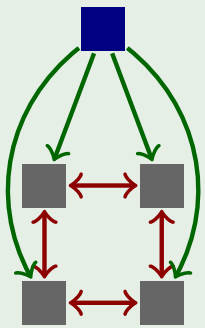


$$\begin{aligned}
\text{Cyclobutane}(x) \quad & \rightarrow G_{cb}(x, f_1(x), f_2(x), f_3(x), f_4(x)) \\
G_{cb}(x, y_1, y_2, y_3, y_4) \quad & \rightarrow \text{Cyclobutane}(x) \wedge \\
& \quad \text{Carbon}(y_1) \wedge \text{Carbon}(y_2) \wedge \\
& \quad \text{Carbon}(y_3) \wedge \text{Carbon}(y_4) \wedge \\
& \quad \text{HasAtom}(x, y_1) \wedge \text{Bond}(y_1, y_2) \wedge \\
& \quad \text{HasAtom}(x, y_2) \wedge \text{Bond}(y_2, y_3) \wedge \\
& \quad \text{HasAtom}(x, y_3) \wedge \text{Bond}(y_3, y_4) \wedge \\
& \quad \text{HasAtom}(x, y_4) \wedge \text{Bond}(y_4, y_1)
\end{aligned}$$

Encoding Description Graphs

- Translate DGs into logic programs with function symbols

Example



$$\begin{aligned}
\text{Cyclobutane}(x) \quad & \rightarrow G_{cb}(x, f_1(x), f_2(x), f_3(x), f_4(x)) \\
G_{cb}(x, y_1, y_2, y_3, y_4) \quad & \rightarrow \text{Cyclobutane}(x) \wedge \\
& \quad \text{Carbon}(y_1) \wedge \text{Carbon}(y_2) \wedge \\
& \quad \text{Carbon}(y_3) \wedge \text{Carbon}(y_4) \wedge \\
& \quad \text{HasAtom}(x, y_1) \wedge \text{Bond}(y_1, y_2) \wedge \\
& \quad \text{HasAtom}(x, y_2) \wedge \text{Bond}(y_2, y_3) \wedge \\
& \quad \text{HasAtom}(x, y_3) \wedge \text{Bond}(y_3, y_4) \wedge \\
& \quad \text{HasAtom}(x, y_4) \wedge \text{Bond}(y_4, y_1)
\end{aligned}$$

- Function symbols allow for schema-level reasoning

Classifying Chemicals

Example

$$\text{Molecule}(x) \wedge \text{HasAtom}(x, y) \wedge \textbf{not}\ \text{Carbon}(y) \wedge \textbf{not}\ \text{Hydrogen}(y) \rightarrow \text{NotHydroCarbon}(x)$$

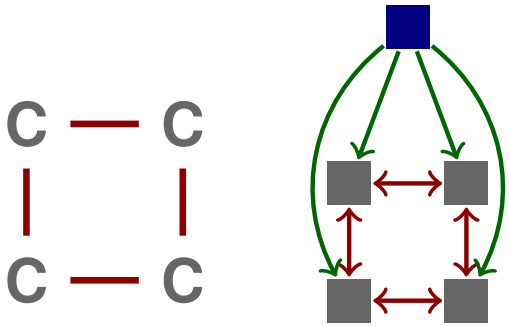
$$\text{Molecule}(x) \wedge \textbf{not}\ \text{NotHydroCarbon}(x) \rightarrow \text{HydroCarbon}(x)$$

Classifying Chemicals

Example

$$\text{Molecule}(x) \wedge \text{HasAtom}(x, y) \wedge \textbf{not}\ \text{Carbon}(y) \wedge \textbf{not}\ \text{Hydrogen}(y) \rightarrow \text{NotHydroCarbon}(x)$$

$$\text{Molecule}(x) \wedge \textbf{not}\ \text{NotHydroCarbon}(x) \rightarrow \text{HydroCarbon}(x)$$

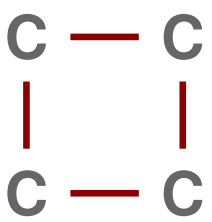


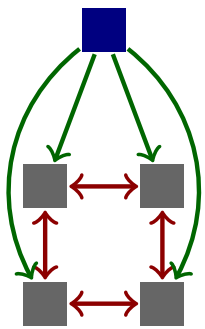
Classifying Chemicals

Example

$$\text{Molecule}(x) \wedge \text{HasAtom}(x, y) \wedge \textbf{not}\ \text{Carbon}(y) \wedge \textbf{not}\ \text{Hydrogen}(y) \rightarrow \text{NotHydroCarbon}(x)$$

$$\text{Molecule}(x) \wedge \textbf{not}\ \text{NotHydroCarbon}(x) \rightarrow \text{HydroCarbon}(x)$$





- Is cyclobutane a hydrocarbon? ✓

Classifying Chemicals

Example

$$\text{Molecule}(x) \wedge \bigwedge_{1 \le i \le 4} \text{HasAtom}(x, y_i) \wedge \bigwedge_{1 \le i \le 3} \text{Bond}(y_i, y_{i+1}) \wedge$$

$$\text{Bond}(y_4, y_1) \bigwedge_{1 \le i < j \le 4} \textbf{not } y_i = y_j$$

$$\rightarrow \text{MoleculeWith4MemberedRing}(x)$$

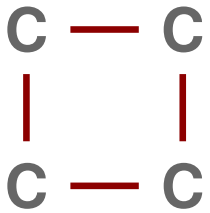
Classifying Chemicals

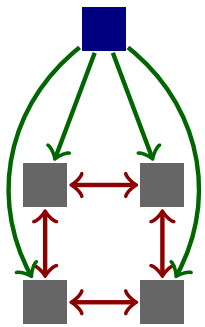
Example

$$\text{Molecule}(x) \wedge \bigwedge_{1 \le i \le 4} \text{HasAtom}(x, y_i) \wedge \bigwedge_{1 \le i \le 3} \text{Bond}(y_i, y_{i+1}) \wedge$$

$$\text{Bond}(y_4, y_1) \bigwedge_{1 \le i < j \le 4} \textbf{not } y_i = y_j$$

$$\rightarrow \text{MoleculeWith4MemberedRing}(x)$$

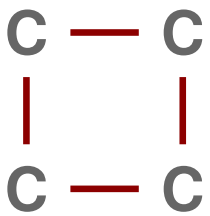


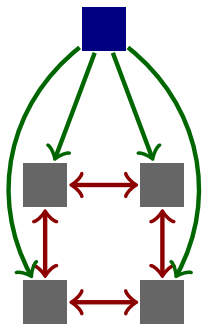


Classifying Chemicals

Example

$$\text{Molecule}(x) \wedge \bigwedge_{1 \le i \le 4} \text{HasAtom}(x, y_i) \wedge \bigwedge_{1 \le i \le 3} \text{Bond}(y_i, y_{i+1}) \wedge$$
$$\text{Bond}(y_4, y_1) \bigwedge_{1 \le i < j \le 4} \textbf{not } y_i = y_j$$
$$\rightarrow \text{MoleculeWith4MemberedRing}(x)$$





- Does cyclobutane contain a four-membered ring? ✓

UNDECIDABILITY

- Logic programs with function symbols can axiomatise infinitely large structures

UNDECIDABILITY

- Logic programs with function symbols can axiomatise infinitely large structures
- Reasoning with DGLP ontologies is trivially undecidable

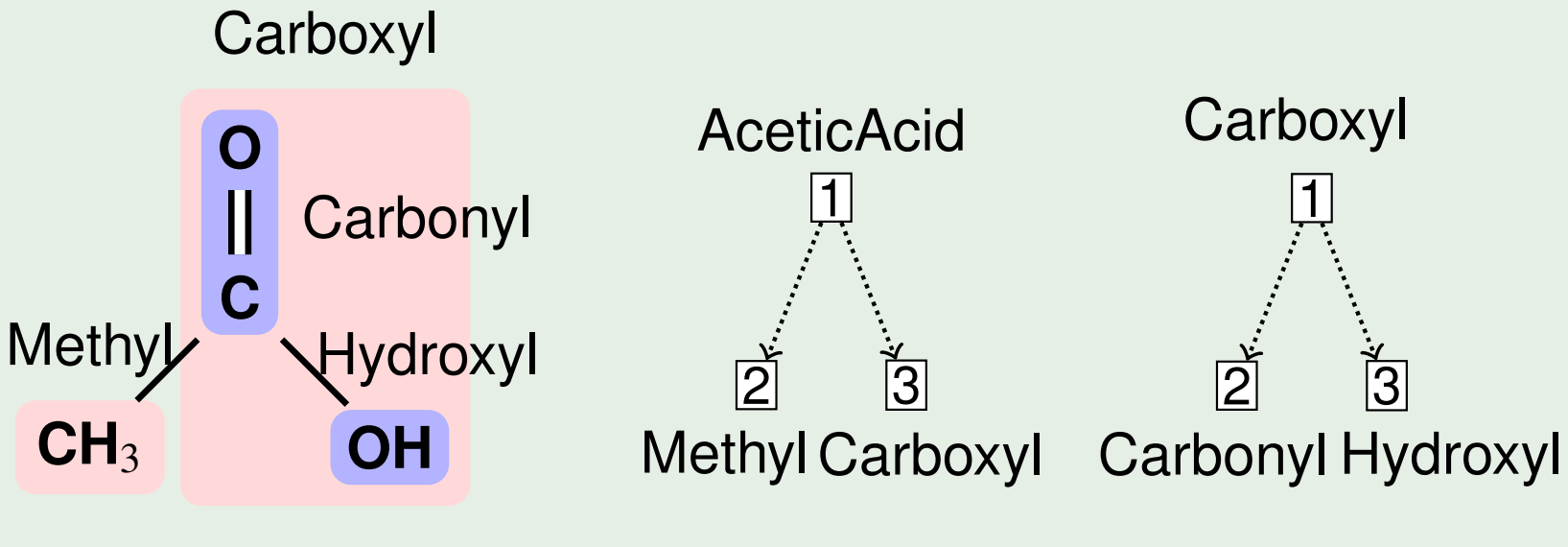
UNDECIDABILITY

- Logic programs with function symbols can axiomatise infinitely large structures
- Reasoning with DGLP ontologies is trivially undecidable
- We are only interested in finite structures

Undecidability

- Logic programs with function symbols can axiomatise infinitely large structures
- Reasoning with DGLP ontologies is trivially undecidable
- We are only interested in finite structures

Example



Acyclicity Conditions

- Formalisms extensively studied, e.g. Datalog$^{\pm}$

Acyclicity Conditions

- Formalisms extensively studied, e.g. Datalog$^{\pm}$
- Various syntax-based acyclicity conditions

Acyclicity Conditions

- Formalisms extensively studied, e.g. Datalog$^{\pm}$
- Various syntax-based acyclicity conditions
  - weak acyclicity [Fagin et al., ICDT, 2002]
  - super-weak acyclicity [Marnette, PODS, 2009]
  - joint acyclicity [Krötzsch and Rudolph, IJCAI, 2011]

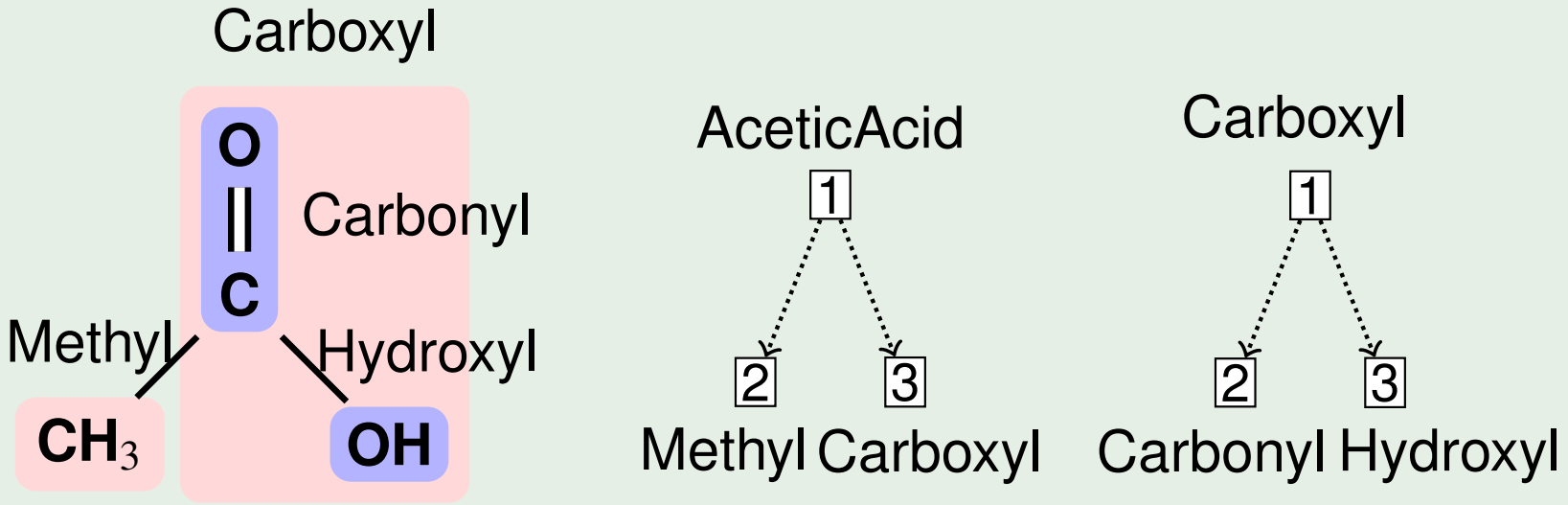
Acyclicity Conditions

- Formalisms extensively studied, e.g. Datalog$^{\pm}$
- Various syntax-based acyclicity conditions
  - weak acyclicity [Fagin et al., ICDT, 2002]
  - super-weak acyclicity [Marnette, PODS, 2009]
  - joint acyclicity [Krötzsch and Rudolph, IJCAI, 2011]
  - rule out naturally-arising nested structures

Acyclicity Conditions

- Formalisms extensively studied, e.g. Datalog$^{\pm}$
- Various syntax-based acyclicity conditions
  - weak acyclicity [Fagin et al., ICDT, 2002]
  - super-weak acyclicity [Marnette, PODS, 2009]
  - joint acyclicity [Krötzsch and Rudolph, IJCAI, 2011]
  - rule out naturally-arising nested structures

Example



Acyclicity Conditions

- Formalisms extensively studied, e.g. Datalog$^{\pm}$
- Various syntax-based acyclicity conditions
  - weak acyclicity [Fagin et al., ICDT, 2002]
  - super-weak acyclicity [Marnette, PODS, 2009]
  - joint acyclicity [Krötzsch and Rudolph, IJCAI, 2011]
  - rule out naturally-arising nested structures
- Novel semantic acyclicity condition

Acyclicity Conditions

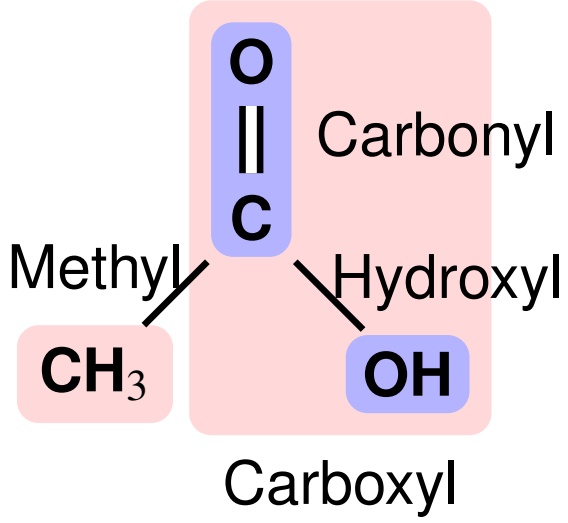
- Formalisms extensively studied, e.g. Datalog$^{\pm}$
- Various syntax-based acyclicity conditions
  - weak acyclicity [Fagin et al., ICDT, 2002]
  - super-weak acyclicity [Marnette, PODS, 2009]
  - joint acyclicity [Krötzsch and Rudolph, IJCAI, 2011]
  - rule out naturally-arising nested structures
- Novel semantic acyclicity condition
  - checks for generation of unbounded structures on the fly

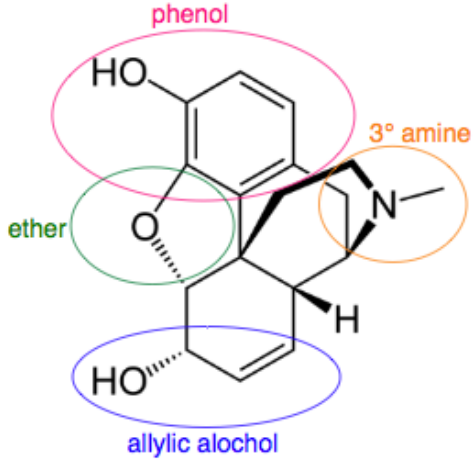
Acyclicity Conditions

- Formalisms extensively studied, e.g. Datalog$^{\pm}$
- Various syntax-based acyclicity conditions
  - weak acyclicity [Fagin et al., ICDT, 2002]
  - super-weak acyclicity [Marnette, PODS, 2009]
  - joint acyclicity [Krötzsch and Rudolph, IJCAI, 2011]
  - rule out naturally-arising nested structures
- Novel semantic acyclicity condition
  - checks for generation of unbounded structures on the fly
  - modelling of molecules that contain functional groups

Acyclicity Conditions

- Formalisms extensively studied, e.g. Datalog$^{\pm}$
- Various syntax-based acyclicity conditions
  - weak acyclicity [Fagin et al., ICDT, 2002]
  - super-weak acyclicity [Marnette, PODS, 2009]
  - joint acyclicity [Krötzsch and Rudolph, IJCAI, 2011]
  - rule out naturally-arising nested structures
- Novel semantic acyclicity condition
  - checks for generation of unbounded structures on the fly
  - modelling of molecules that contain functional groups





Empirical Evaluation

- Data extracted from ChEBI in Molfile format

Empirical Evaluation

- Data extracted from ChEBI in Molfile format
- XSB logic programming engine

Empirical Evaluation

- Data extracted from ChEBI in Molfile format
- XSB logic programming engine
- Chemical classes:

Empirical Evaluation

- Data extracted from ChEBI in Molfile format
- XSB logic programming engine
- Chemical classes:
  - Hydrocarbons
  - Inorganic molecules
  - Molecules with exactly two carbons
  - Molecules with a four-membered ring
  - Molecules with a benzene

Empirical Evaluation

- Data extracted from ChEBI in Molfile format
- XSB logic programming engine
- Chemical classes:
  - Hydrocarbons
  - Inorganic molecules
  - Molecules with exactly two carbons
  - Molecules with a four-membered ring
  - Molecules with a benzene
- Preliminary evaluation ranging from 10 to 70 molecules

Empirical Evaluation

- Data extracted from ChEBI in Molfile format
- XSB logic programming engine
- Chemical classes:
  - Hydrocarbons
  - Inorganic molecules
  - Molecules with exactly two carbons
  - Molecules with a four-membered ring
  - Molecules with a benzene
- Preliminary evaluation ranging from 10 to 70 molecules
- Results:

Empirical Evaluation

- Data extracted from ChEBI in Molfile format
- XSB logic programming engine
- Chemical classes:
  - Hydrocarbons
  - Inorganic molecules
  - Molecules with exactly two carbons
  - Molecules with a four-membered ring
  - Molecules with a benzene
- Preliminary evaluation ranging from 10 to 70 molecules
- Results:
  - All DGLP ontologies were found acyclic

Empirical Evaluation

- Data extracted from ChEBI in Molfile format
- XSB logic programming engine
- Chemical classes:
  - Hydrocarbons
  - Inorganic molecules
  - Molecules with exactly two carbons
  - Molecules with a four-membered ring
  - Molecules with a benzene
- Preliminary evaluation ranging from 10 to 70 molecules
- Results:
  - All DGLP ontologies were found acyclic
  - Molecules classified as expected

Empirical Evaluation

- Data extracted from ChEBI in Molfile format
- XSB logic programming engine
- Chemical classes:
  - Hydrocarbons
  - Inorganic molecules
  - Molecules with exactly two carbons
  - Molecules with a four-membered ring
  - Molecules with a benzene
- Preliminary evaluation ranging from 10 to 70 molecules
- Results:
  - All DGLP ontologies were found acyclic
  - Molecules classified as expected
  - Suite of subsumption tests for largest ontology performed in few minutes

Overview and Future Directions

1. Expressive and decidable formalism for representation of structured objects

Overview and Future Directions

1. Expressive and decidable formalism for representation of structured objects
2. Novel acyclicity condition for logic programs with restricted use of function symbols

Overview and Future Directions

1. Expressive and decidable formalism for representation of structured objects
2. Novel acyclicity condition for logic programs with restricted use of function symbols
3. Prototype for the automated classification of chemicals

Overview and Future Directions

1. Expressive and decidable formalism for representation of structured objects
2. Novel acyclicity condition for logic programs with restricted use of function symbols
3. Prototype for the automated classification of chemicals
   - Is dinitrogen inorganic?
   - Does cyclobutane contain a four-membered ring?
   - Is acetylene a hydrocarbon?
   - Does benzaldehyde contain a benzene ring?

Overview and Future Directions

1. Expressive and decidable formalism for representation of structured objects
2. Novel acyclicity condition for logic programs with restricted use of function symbols
3. Prototype for the automated classification of chemicals
   - Is dinitrogen inorganic? ✓
   - Does cyclobutane contain a four-membered ring? ✓
   - Is acetylene a hydrocarbon? ✓
   - Does benzaldehyde contain a benzene ring? ✓

Overview and Future Directions

1. Expressive and decidable formalism for representation of structured objects
2. Novel acyclicity condition for logic programs with restricted use of function symbols
3. Prototype for the automated classification of chemicals
   - Is dinitrogen inorganic? ✓
   - Does cyclobutane contain a four-membered ring? ✓
   - Is acetylene a hydrocarbon? ✓
   - Does benzaldehyde contain a benzene ring? ✓

- Future directions:

Overview and Future Directions

1. Expressive and decidable formalism for representation of structured objects
2. Novel acyclicity condition for logic programs with restricted use of function symbols
3. Prototype for the automated classification of chemicals
   - Is dinitrogen inorganic? ✓
   - Does cyclobutane contain a four-membered ring? ✓
   - Is acetylene a hydrocarbon? ✓
   - Does benzaldehyde contain a benzene ring? ✓

- Future directions:
  - Datalog rules with existentials in the head

Overview and Future Directions

1. Expressive and decidable formalism for representation of structured objects
2. Novel acyclicity condition for logic programs with restricted use of function symbols
3. Prototype for the automated classification of chemicals
   - Is dinitrogen inorganic? ✓
   - Does cyclobutane contain a four-membered ring? ✓
   - Is acetylene a hydrocarbon? ✓
   - Does benzaldehyde contain a benzene ring? ✓

- Future directions:
  - Datalog rules with existentials in the head
  - User-friendly surface syntax

Overview and Future Directions

1. Expressive and decidable formalism for representation of structured objects
2. Novel acyclicity condition for logic programs with restricted use of function symbols
3. Prototype for the automated classification of chemicals
   - Is dinitrogen inorganic? ✓
   - Does cyclobutane contain a four-membered ring? ✓
   - Is acetylene a hydrocarbon? ✓
   - Does benzaldehyde contain a benzene ring? ✓

- Future directions:
  - Datalog rules with existentials in the head
  - User-friendly surface syntax
  - Fully-fledged chemical classification system

Overview and Future Directions

1. Expressive and decidable formalism for representation of structured objects
2. Novel acyclicity condition for logic programs with restricted use of function symbols
3. Prototype for the automated classification of chemicals
   - Is dinitrogen inorganic? ✓
   - Does cyclobutane contain a four-membered ring? ✓
   - Is acetylene a hydrocarbon? ✓
   - Does benzaldehyde contain a benzene ring? ✓

- Future directions:
  - Datalog rules with existentials in the head
  - User-friendly surface syntax
  - Fully-fledged chemical classification system
- Thank you for listening. Questions?