Declarative Problem Solving and Nonmonotonic Reasoning

Thomas Eiter

Institute of Information Systems
Vienna University of Technology
eiter@kr.tuwien.ac.at

Japan-Austria Joint Workshop on ICT, Tokyo, Oct 18-19, 2010

TECHNISCHE UNIVERSITÄT WIEN
Vienna University of Technology

http://www.tuwien.ac.at/
http://www.cs.tuwien.ac.at/

Facts:

- Established 1815
- Currently, about 150 full professors and 1800 scientific staff, plus 600 teaching assistants, 24,000 students
- 8 faculties, including *Faculty of Informatics*
- Faculty of Informatics has 7 institutes (currently 20+ full profs, 35+ associate profs); since 2009/10 a *PhD School*
- Affinity to Knowledge Engineering and IS: about 16 profs

Institute of Information Systems

http://www.informatik.tuwien.ac.at/institute/e184.html

- One of the largest institutes in the Faculty of Informatics
- Four groups
  - **Distributed Systems Group (DSG)**
    Profs. Dustdar, N.N.
  - **Databases and AI (DBAI)**
    Profs. Pichler, Gottlob
  - **Knowledge Based Systems Group (KBS)**
    Profs. Eiter, Szeider
  - **Formal Methods in Systems Engineering (FORSYTE)**
    Prof. Veith
- Personal: $\approx$ 70 scientific staff, $\approx$ 10 administrative/technical staff
- Head: Prof. Eiter

Projects

International Projects

- EU Projects (FPx)
  - Networks of Excellence
    (CologNet, REWERSE, S-CUBE, GAMES, MONET,...)
  - Integrated Projects, Streps
    (Ontorule, INFOMIX, SM4ALL, COMPAS, COIN, COMMIUS, NEDINE,...)
  - Erasmus Mundus: European Master in Computational Logic
  - IRSES (Net2)
- Bilateral projects
- ESA

National Projects

- **FWF**
- FFG (FIT-IT Line, ...)
- WWTF (INCMAN, SODI, ARGUMENTATION, FOS)
- ÖAW (Doc)

Distributed Systems Group (DSG)

http://www.infosys.tuwien.ac.at/

Profs. S. Dustdar, N.N.

- Software architectures
- Software services and components
- Distributed services
  - Foundations of Service-oriented Computing
  - Autonomic, Complex, and Context-aware Computing
  - Grid Computing
  - Mobile and Ubiquitous Computing
- Novel paradigms for distributed systems

Databases and Artificial Intelligence Group (DBAI)

http://www.dbai.tuwien.ac.at/

Profs. G. Gottlob (Oxford University), R. Pichler, S. Woltran

- Foundations of databases
- Computational logic and complexity
- Semi-structured data
- Advanced database systems
  - data integration, data exchange
- Web data and information extraction
  - Spin-Off: http://www.lixto.com/

Tools & middleware for visual data wrapper construction

Knowledge Based Systems Group (KBS)

`http://www.kr.tuwien.ac.at/`

Profs. U. Egly, T. Eiter, S. Szeider, H. Tompits

- Computational logic and complexity
  - SAT/QBF solving, theorem proving, discrete methods
  - DLV + extensions (DLVHEX, dl-programs, . . .)
- **Knowledge representation and reasoning**
  - Inconsistency management
  - Contextual reasoning
  - Action languages and agents (DLV^{K}, IMPACT)
  - Ontologies, Description Logics
- **Declarative problem solving**
  - *Answer Set Programming* (ASP)
- Mobile robots
- KBS in engineering

Nonmonotonic Reasoning

- Classical Logic (propositional logic, first-order logic, modal logic) has the property of monotonicity:

  If $T \vdash \phi$ and $T \subseteq T'$, then $T' \vdash \phi$

  That is, a conclusion remains valid if new sentences are added to T.

Nonmonotonic Reasoning

- Classical Logic (propositional logic, first-order logic, modal logic) has the property of monotonicity:

  If $T \vdash \phi$ and $T \subseteq T'$, then $T' \vdash \phi$

  That is, a conclusion remains valid if new sentences are added to T.
- Common-sense reasoning is typically *nonmonotonic*.
  That is, from $T' \vdash \phi$ might not hold.
- One reason for this is that humans must draw conclusions in situations of *incomplete information*.
- While classical logic remains agnostic in such a situation, common-sense reasoning is based on *reasonable assumptions*.

Example

KB = { (1) $\forall x. french\_guy(x) \wedge \neg mute(x) \Rightarrow speaks\_french(x)$
"Non-mute French guys speak French."
(2) $\forall x. mute(x) \Rightarrow \neg speaks\_french(x)$
"Mute persons do not speak French."
(3) $french\_guy(luc)$
"Luc is a french guy." }

- Does $\mathbf{KB} \vdash speaks\_french(luc)$?

Example

KB = { (1) $\forall x.french\_guy(x) \wedge \neg mute(x) \Rightarrow speaks\_french(x)$
"Non-mute French guys speak French."
(2) $\forall x.mute(x) \Rightarrow \neg speaks\_french(x)$
"Mute persons do not speak French."
(3) $french\_guy(luc)$
"Luc is a french guy." }

- Does $\mathbf{KB} \vdash speaks\_french(luc)$?
  - Classical Logic: $\mathbf{KB} \nvdash speaks\_french(luc)$

Example

KB = { (1) $\forall x.french\_guy(x) \land \neg mute(x) \Rightarrow speaks\_french(x)$
"Non-mute French guys speak French."
(2) $\forall x.mute(x) \Rightarrow \neg speaks\_french(x)$
"Mute persons do not speak French."
(3) $french\_guy(luc)$
"Luc is a french guy." }

- Does $\mathbf{KB} \vdash speaks\_french(luc)$?
  - Classical Logic: $\mathbf{KB} \not\vdash speaks\_french(luc)$
  - Commonsense Reasoning: conclude $speaks\_french(luc)$.

Example

KB = { (1) $\forall x.french\_guy(x) \wedge \neg mute(x) \Rightarrow speaks\_french(x)$
"Non-mute French guys speak French."
(2) $\forall x.mute(x) \Rightarrow \neg speaks\_french(x)$
"Mute persons do not speak French."
(3) $french\_guy(luc)$
"Luc is a french guy." }

- Does $\mathbf{KB} \vdash speaks\_french(luc)$?
  - Classical Logic: $\mathbf{KB} \not\vdash speaks\_french(luc)$
  - Commonsense Reasoning: conclude $speaks\_french(luc)$.
- Add new information: $mute(luc)$

Example

KB = { (1) $\forall x. \mathit{french\_guy}(x) \land \neg \mathit{mute}(x) \Rightarrow \mathit{speaks\_french}(x)$
"Non-mute French guys speak French."
(2) $\forall x. \mathit{mute}(x) \Rightarrow \neg \mathit{speaks\_french}(x)$
"Mute persons do not speak French."
(3) $\mathit{french\_guy}(\mathit{luc})$
"Luc is a french guy." }

- Does $\mathbf{KB} \vdash \mathit{speaks\_french}(\mathit{luc})$?
  - Classical Logic: $\mathbf{KB} \nvdash \mathit{speaks\_french}(\mathit{luc})$
  - Commonsense Reasoning: conclude $\mathit{speaks\_french}(\mathit{luc})$.
- Add new information: $\mathit{mute}(\mathit{luc})$
  - In both classical logic and commonsense reasoning: conclude $\neg \mathit{speaks\_french}(\mathit{luc})$, but not $\mathit{speaks\_french}(\mathit{luc})$.

Nonmonotonic Formalisms

- Default Logic (Reiter 1980)
- Nonmonotonic Logic (NML, McDermott & Doyle 1980)
- Autoepistemic Logic (R. Moore 1985)
- Abductive Reasoning (C.S.Peirce; Selman & Levesque 1990, Bylander 1991)
- **Extended Logic Programs (Gelfond & Lifschitz 1991)**
  A rule based formalism, can be viewed as fragment of Default Logic

$$\mathbf{P} = \{ \begin{array}{l} speaks\_french(x) :- french\_guy(x), \mathrm{not}\, mute(x). \\ \neg speaks\_french(x) :- mute(x). \\ french\_guy(luc). \end{array} \}$$

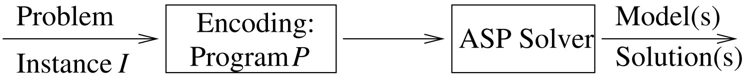
Basis for the ***Answer Set Programming Paradigm***

Answer Set Programming (ASP)

A recent *declarative problem solving method*

General idea

Reduce solving of a problem I to computing models of a logic program / SAT theory



1. *Encode* I as a (non-monotonic) logic program P, such that solutions of I are represented by models of P
2. *Compute* some model M of P, using an ASP solver
3. *Extract* some solution for I from M.

Example: Graph 3-Coloring

Color all nodes of a graph with colors r, g, b such that adjacent nodes have different color.

Problem specification P_{PS}

$$
\left.\begin{array}{r} g(X) \lor r(X) \lor b(X) \leftarrow node(X) \end{array}\right\} \textbf{Guess}
$$

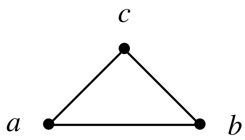
$$
\left.\begin{array}{l} \leftarrow b(X), b(Y), edge(X,Y) \\ \leftarrow r(X), r(Y), edge(X,Y) \\ \leftarrow g(X), g(Y), edge(X,Y) \end{array}\right\} \textbf{Check}
$$

Data P_D: Graph $G = (V, E)$

$$
P_D = \{node(v) \mid v \in V\} \cup \{edge(v,w) \mid (v,w) \in E\}.
$$

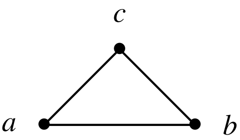
3-colorings $\rightleftharpoons$ models:
$v \in V$ has color $c \in \{r, g, b\}$ iff $c(v)$ is in the corr. model of $P_{PS} \cup P_D$.

Example: 3-Coloring (ctd.)

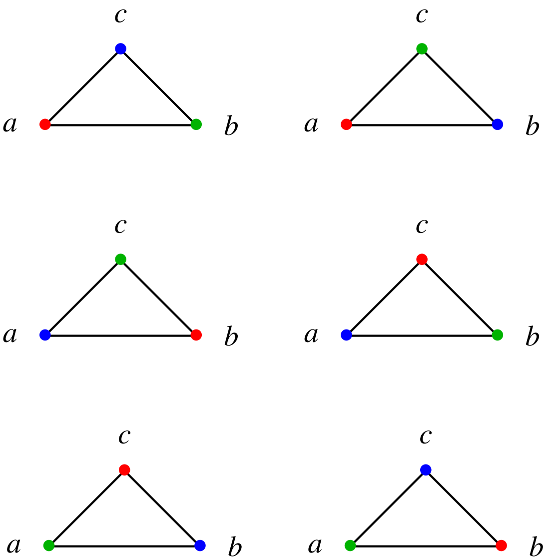


$$P_D = \{node(a), node(b), \\ node(c), edge(a,b), \\ edge(b,c), edge(a,c)\}$$

Example: 3-Coloring (ctd.)



$$P_D = \{node(a), node(b),\\ node(c), edge(a,b),\\ edge(b,c), edge(a,c)\}$$



ASP Applications

Problems in many domains, see
`http://www.kr.tuwien.ac.at/projects/WASP/report.html`

- configuration
- planning, routing
- diagnosis (E.g., Space shuttle reaction control)
- security analysis
- verification
- bioinformatics
- knowledge management
- musicology
- ...

ASP Showcase:
`http://www.kr.tuwien.ac.at/projects/WASP/showcase.html`

DLV System (TU Wien / Università della Calabria)

- DLV is a state-of-the-art disjunctive Answer Set solver
- Based on strong theoretical foundations
- Many constructs ($\Rightarrow$ high expressivness)

$$works(X) :- component(X), \text{not}\, broken(X).$$
$$male(X) \vee female(X) :- person(X).$$

  - non-monotonic negation
  - nondeterministic choice (disjunction)
  - soft / weight constraints
  - aggregates

- Front-ends for specific problems (diagnosis, planning, etc.).
- Extensions: DLVHEX, DLV^{DB}, DLV-Complex, dl-programs, OntoDLV, ...,
- Industrial applications: Exeura Srl `www.exeura.it/`

Ongoing Work and Projects

- Software Engineering for ASP (FWF)
  - Tools, debugging, methodologies
- Modular hex-programs (FWF)
  - hex-programs: extend logic programs with API to access external software
  - Systems of logics programs / modular composition
- Open answer set programming (FWF)

Theory, prototypes, applications

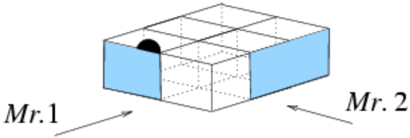
Future work and topics for collaboration

- Deployment of declarative and tools to innovative applications
    - In particular, ASP + extensions, MCS

  Example: personalization
    - Project myITS (customized intelligent travel assistant service)
- Development of domain specific reasoning languages
- Foundations of reasoning (semantics, complexity, algorithms)
    - modular ASP, distributed algorithms
    - inconsistency management
    - ontology reasoning
- Systems
    - DLVHEX++, DMCS , ...

Contextual Reasoning

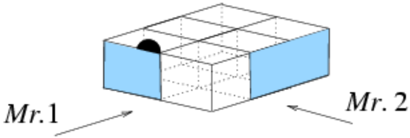
Magic Box



- J. McCarthy: How to interrelate contexts?

Contextual Reasoning

Magic Box

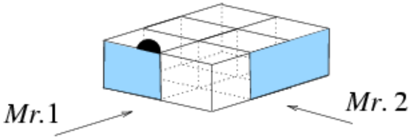


- J. McCarthy: How to interrelate contexts?
- Trento School (Giunchiglia, Serafini et al.)
  Bridge rules for information exchange

$$\begin{array}{ll} Mr.1: & row(X) \leftarrow (Mr.2, sees\_row(X)) \\ Mr.2: & col(Y) \leftarrow (Mr.1, sees\_col(Y)) \end{array}$$

Contextual Reasoning

Magic Box



- J. McCarthy: How to interrelate contexts?
- Trento School (Giunchiglia, Serafini et al.)
  Bridge rules for information exchange

$$
\begin{array}{ll}
Mr.1: & row(X) \leftarrow (Mr.2, sees\_row(X)) \\
Mr.2: & col(Y) \leftarrow (Mr.1, sees\_col(Y))
\end{array}
$$

- Brewka & E_: *Nonmonotonic Multi Context Systems* (MCS)

Nonmonotonic Multi-Context Systems

$$M = (C_1, \ldots, C_n)$$

consists of contexts

$$C_i = (L_i, kb_i, br_i),\ i = 1, \ldots, n,$$

where

- each L_i is an (abstract) "logic,"
- each $kb_i \in \mathbf{KB}_i$ is a knowledge base in L_i, and
- each br_i is a set of bridge rules (possibly with negation)

- Captures many popular logics L_i, e.g. description logics, modal logics, temporal logics, default logics, logic programs
- Semantics in terms of *equilibria*, which are stable states $S = (S_1, \ldots, S_n)$ of M evaluating the kb_i and br_i

Example

Suppose a MCS $M = (C_1, C_2)$ has two contexts, expressing the individual views of a paper by its authors.

- C_1:
  - L_1 = Classical Logic
  - $kb_1 = \{\, unhappy \supset revision \,\}$
  - $br_1 = \{\ unhappy \leftarrow (2 : work)\ \}$
- C_2:
  - L_2 = Default Logic (R.Reiter)
  - $kb_2 = \{\, good : accepted/accepted \,\}$
  - $br_2 = \{\ work \leftarrow (1 : revision),$
    $good \leftarrow \mathrm{not}(1 : unhappy)\ \}$

M has two equilibria:

- $E_1 = (Cn(\{unhappy, revision\}), Cn(\{work\}))$ and
- $E_2 = (Cn(\{unhappy \supset revision\}), Cn(\{good, accepted\}))$

MCS Features

- A rich framework for interlinking heterogeneous knowledge systems
- Fixpoint characterizations (under operational semantics)
- Relationship to game-theoretic concepts (e.g., Nash-equilibria of particular games, sometimes)

Ongoing work and projects

- Algorithms: distributed evaluation (DMCS system prototype)
- WWTF Project *Inconsistency Management for Knowledge-Integration Systems*
  - a general formalism and a suite of basic methods for inconsistency management in MCS,
  - algorithms for their practical realization.
- Special purpose MCS, e.g., in the context of argumentation

Reasoning in Ontologies

- Formal ontologies serve for making conceptual models of domains (human anatomy, airplanes, products, ...)
- *Description Logics* are the premier logic-based formalism for ontology representation.
- They model *concepts* (classes of objects) and *roles* (binary relations between objects).
- A *DL knowledge base* comprises a *taxonomoy part* (*T-Box*) and *assertions* (*A-Box*, facts).

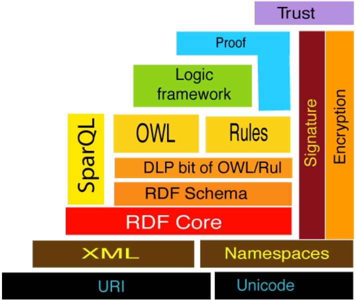
Example: Genealogy

$$\text{T-Box} = \left\{ \begin{array}{l} Person \equiv Female \sqcup Male, \\ Parent \equiv \exists hasChild.Person, \\ HasNoSons \equiv Parent \sqcap \forall hasChild.Female \end{array} \right\}$$

$$\text{A-Box} = \{Parent(Mary), hasChild(Tom, Jen), Female(Jen)\}$$

Applications

- DLs find increasing importance, e.g., for
  - data integration
  - peer-to-peer data management
  - *Semantic Web*



- The Web Ontology Language (OWL 1 / 2) is W3C standard which *builds on Description Logics*

Beyond Ontologies

- DL ontologies have limited expressiveness (OWL 1 $\rightarrow$ OWL 2)
  - constraints ("every person has a SSN")
  - rules ("male siblings of a parent are uncles")
  - combine with traditional databases
  - **mismatch**: Unique Names, Open/Closed World Assumption

supplier	branch	address
Barilla	Roma	Piazza Espagna 1
DeCecco	Milano	Via Cadorno 2
Barilla	Roma	Via Salaria 10

- *dl-programs* bridge the gap: couple ASP and DL via *query atoms*
- **Ongoing projects**:
  - ONTORULE: Ontologies meet Business Rules (ICT FP7)
    (10 partners, including ILOG/IBM, AUDI, ArcelorMittal, OntoPrise)
    (FP7 Ontorule)
  - Reasoning in Hybrid Knowledge Bases (FWF)